

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 561 609**

51 Int. Cl.:

H04N 19/70 (2014.01)

H04N 19/61 (2014.01)

H04N 19/103 (2014.01)

H04N 19/463 (2014.01)

H04N 19/91 (2014.01)

H04N 21/236 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **05.04.2013 E 13717669 (9)**

97 Fecha y número de publicación de la concesión europea: **04.11.2015 EP 2847997**

54 Título: **Agrupación de elementos sintácticos codificados por circunvalación en la codificación de vídeo**

30 Prioridad:

11.04.2012 US 201261623004 P

27.04.2012 US 201261639836 P

15.03.2013 US 201313839855

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
29.02.2016

73 Titular/es:

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

72 Inventor/es:

**CHIEN, WEI-JUNG;
CHEN, JIANLE;
COBAN, MUHAMMED ZEYD y
KARCZEWICZ, MARTA**

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 561 609 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

Agrupación de elementos sintácticos codificados por circunvalación en la codificación de vídeo

5 SOLICITUDES RELACIONADAS

CAMPO TÉCNICO

10 Esta divulgación se refiere a la codificación de vídeo y, más en particular, a técnicas para codificar elementos sintácticos en la codificación de vídeo.

ANTECEDENTES

15 Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisiones digitales, sistemas de radiodifusión directa digitales, sistemas de radiodifusión inalámbricos, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, cámaras digitales, dispositivos de grabación digitales, reproductores multimedia digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados teléfonos "inteligentes", ordenadores tipo tableta, dispositivos de videoconferencia, etc. Las normas de codificación de vídeo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo multivista (MVC). Además, la codificación de vídeo de alta eficiencia (HEVC) es una norma de codificación de vídeo que está desarrollándose 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.

30 Las técnicas de compresión de vídeo llevan a cabo una predicción espacial y/o una predicción temporal para reducir o eliminar la redundancia presente en las secuencias de vídeo. En la codificación de vídeo predictiva basada en bloques, un cuadro de vídeo o un fragmento de un cuadro de vídeo puede dividirse en bloques de vídeo, que pueden denominarse macrobloques o unidades de codificación. Cada bloque de vídeo puede dividirse adicionalmente. Los bloques de vídeo de un cuadro o fragmento intra-codificado (I) se codifican usando predicción espacial con respecto a bloques de vídeo vecinos. Los bloques de vídeo de un cuadro o fragmento inter-codificado (P o B) pueden usar predicción espacial con respecto a bloques de vídeo vecinos del mismo cuadro o fragmento, o predicción temporal con respecto a otros cuadros de referencia.

40 Un bloque inter-codificado puede codificarse usando un vector de movimiento que indica un grupo de muestras de referencia usadas para formar un bloque predictivo. Un bloque intra-codificado puede codificarse usando un intra-modo que apunta a un bloque de muestras de referencia usadas para formar un bloque predictivo. La diferencia entre el bloque de vídeo y el bloque predictivo puede denominarse datos residuales. Para obtener una mayor compresión, los datos residuales pueden transformarse desde el dominio de píxeles a un dominio de transformada, dando como resultado coeficientes de transformada residuales, los cuales pueden cuantificarse posteriormente. Además, los coeficientes de transformada cuantificados y la información predictiva pueden codificarse por entropía para una mayor compresión.

50 El documento de FRANCOIS E ET AL: "*CE6b: Intra mode coding with 4 MPMs and mode ranking*", 7. CONFERENCIA DE JCT-VC ; 98. CONFERENCIA DE MPEG; del 21/11/2011 al 30/11/2011; GINEBRA; (EQUIPO DE COLABORACIÓN CONJUNTA EN CODIFICACIÓN DE VÍDEO DE ISO/IEC JTC1/SC29/WG11 E ITU-T SG.16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/](http://wftp3.itu.int/av-arch/jctvc-site/) no. JCTVC-G243, 8 de noviembre de 2011 (08/11/2011), XP030110227, da a conocer una solución de codificación intra-modo relacionada con el uso de modos más probables adicionales y el uso de clasificación de modo para la codificación de modo restante. En particular, se da a conocer el uso de cuatro MPM (2 MPM convencionales + 2 MPRM) y la clasificación de modo para codificar *celdas* de modo restantes. Las mismas tablas VLC y de binarización se unifican en las configuraciones LC y HE. En HE se usa codificación por circunvalación en lugar de CABAC para los MPRM y los de modo restantes. La codificación de crominancia también se modifica considerando en los conjuntos candidatos los 4 MPM de luminancia. Se añade un décimo noveno modo al conjunto de modos de bloques 4x4 para obtener un conjunto más coherente de modos.

60 El documento de KARCZEWICZ M ET AL: "*Improvements on VLC*", 3. CONFERENCIA DE JCT-VC ; 94. CONFERENCIA DE MPEG; del 07/10/2010 al 15/10/2010; GUANGZHOU; (EQUIPO DE COLABORACIÓN CONJUNTA EN CODIFICACIÓN DE VÍDEO DE ISO/IEC JTC1/SC29/WG11 E ITU-T SG.16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/](http://wftp3.itu.int/av-arch/jctvc-site/) no. JCTVC-C263, 3 de octubre de 2010 (03/10/2010), XP030007970, ISSN: 0000-0019, se refiere a la codificación de longitud variable (VLC) y, en particular, a una modificación de la codificación actual de coeficientes de transformada para los intra-bloques. Se

sugiere que las nuevas tablas de correlación VLC y que los cambios en la selección de tablas en el par de codificación (*levelID, run*) pueden conseguir una ganancia de codificación del 2,3 %. El documento también propone nuevos esquemas de codificación para el modo de predicción de los intra-bloques y los inter-bloques, respectivamente.

5

RESUMEN

En general, esta divulgación describe varias técnicas para ordenar elementos sintácticos en la codificación de vídeo predictiva. Estas técnicas pueden reducir la información de control en las cadenas de procesamiento, la cual puede afectar al rendimiento de la codificación de entropía. En un ejemplo, el orden en que los elementos sintácticos se codifican mediante CABAC (codificación aritmética binaria adaptable al contexto) o se codifican por circunvalación puede optimizarse. La codificación por circunvalación puede referirse a la codificación aritmética de *celdas* sin usar un contexto adaptativo.

10

En un ejemplo, los elementos sintácticos que indican modos de intra-predicción de unidades predictivas (PU) en una unidad de codificación (CU) pueden agruparse juntos dependiendo de si van a codificarse por circunvalación o codificarse mediante CABAC. Esta agrupación puede reducir el número total de ciclos requeridos para codificar por entropía los elementos sintácticos.

15

En un ejemplo, la divulgación describe un procedimiento de descodificación de datos de vídeo, comprendiendo el procedimiento recibir un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos y un segundo grupo de elementos sintácticos para las unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva de la CU está basado en un índice hacia una lista de modos más probables; descodificar según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de descodificar mediante CABAC el primer grupo de elementos sintácticos, descodificar por circunvalación el segundo grupo de elementos sintácticos; y reconstruir los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

20

25

30

En otro ejemplo, la divulgación describe un aparato de descodificación de datos de vídeo, comprendiendo el aparato un descodificador de vídeo configurado para recibir un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos y un segundo grupo de elementos sintácticos para las unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva de la CU está basado en un índice hacia una lista de modos más probables; descodificar según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de descodificar mediante CABAC el primer grupo de elementos sintácticos, descodificar por circunvalación el segundo grupo de elementos sintácticos; y reconstruir los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

35

40

45

En otro ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo de descodificación de datos de vídeo reciban un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos y un segundo grupo de elementos sintácticos para las unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva de la CU está basado en un índice hacia una lista de modos más probables; descodifiquen según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; descodifiquen por circunvalación el segundo grupo de elementos sintácticos, después de descodificar mediante CABAC el primer grupo de elementos sintácticos; y reconstruyan los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

50

55

En otro ejemplo, la divulgación describe un aparato de descodificación de datos de vídeo, comprendiendo el aparato medios para recibir un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos y un segundo grupo de elementos sintácticos para las unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva de la CU está basado en un índice hacia una lista de modos más

60

65

probables; medios para descodificar según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; medios para descodificar por circunvalación el segundo grupo de elementos sintácticos, después de descodificar mediante CABAC el primer grupo de elementos sintácticos; y medios para reconstruir los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

En otro ejemplo, la divulgación describe un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento generar un primer grupo de elementos sintácticos, donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una unidad de predicción respectiva (PU) de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; generar un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos; codificar mediante codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de codificar mediante CABAC el primer grupo de elementos sintácticos, codificar por circunvalación el segundo grupo de elementos sintácticos; y proporcionar los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.

En otro ejemplo, la divulgación describe un aparato de codificación de datos de vídeo, comprendiendo el aparato un codificador de vídeo configurado para generar un primer grupo de elementos sintácticos, donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una unidad de predicción respectiva (PU) de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; generar un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos; codificar mediante codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de codificar mediante CABAC el primer grupo de elementos sintácticos, codificar por circunvalación el segundo grupo de elementos sintácticos; y proporcionar los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.

En otro ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo de codificación de datos de vídeo generen un primer grupo de elementos sintácticos, donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una unidad de predicción respectiva (PU) de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; generen un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos; codifiquen mediante codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; codifiquen por circunvalación el segundo grupo de elementos sintácticos, después de codificar mediante CABAC el primer grupo de elementos sintácticos; y proporcionen los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.

En otro ejemplo, la divulgación describe un aparato de codificación de datos de vídeo, comprendiendo el aparato medios para generar un primer grupo de elementos sintácticos, donde un elemento sintáctico del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una unidad de predicción respectiva (PU) de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; medios para generar un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos; medios para codificar mediante codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; medios para codificar por circunvalación el segundo grupo de elementos sintácticos, después de codificar mediante CABAC el primer grupo de elementos sintácticos; y medios para proporcionar los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.

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, los dibujos y las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

La FIG. 1 es un diagrama conceptual que ilustra ejemplos de estructuras de unidades de predicción y modos de codificación de intra-predicción asociados.

La FIG. 2 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo de ejemplo, que puede estar configurado para utilizar técnicas según esta divulgación.

La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas para codificar elementos sintácticos predictivos, según esta divulgación.

5 La FIG. 4 es un diagrama de bloques que ilustra un ejemplo de un codificador de entropía que puede implementar técnicas para codificar por entropía elementos sintácticos predictivos, según esta divulgación.

10 La FIG. 5 es un diagrama de flujo que ilustra elementos sintácticos predictivos de codificación según las técnicas de esta divulgación.

15 La FIG. 6 es un diagrama de bloques que ilustra un ejemplo de un descodificador de vídeo que puede implementar técnicas para descodificar una secuencia de vídeo codificada que incluye elementos sintácticos predictivos, según esta divulgación.

La FIG. 7 es un diagrama de bloques que ilustra un ejemplo de un descodificador de entropía que puede implementar técnicas para descodificar por entropía elementos sintácticos predictivos, según esta divulgación.

20 La FIG. 8 es un diagrama de flujo que ilustra elementos sintácticos predictivos de descodificación según las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

25 Los elementos sintácticos predictivos de un grupo de elementos sintácticos, tal como elementos sintácticos que pueden indicar un modo de intra-predicción para una o más unidades de predicción, pueden codificarse mediante codificación aritmética binaria adaptable al contexto (CABAC) convencional o codificarse por circunvalación. El orden en que estos elementos sintácticos predictivos se reciben en un codificador CABAC puede afectar al número total de ciclos de procesamiento requeridos para codificar por entropía el grupo de
30 elementos sintácticos. Esto puede deberse a la información de configuración requerida para volver a activar un motor de codificación CABAC en las transiciones de codificación entre la codificación por circunvalación y la codificación CABAC. En las técnicas descritas en esta divulgación, estos elementos sintácticos pueden ordenarse de tal manera que se reduzca el número total de ciclos de procesamiento requeridos para codificar por entropía el grupo de elementos sintácticos.

35 Por ejemplo, en otras técnicas, en lo que respecta a la intra-predicción, un codificador de vídeo señala elementos sintácticos que especifican el modo de intra-predicción de muestras de luminancia y señala elementos sintácticos para las muestras de crominancia. Algunos de los elementos sintácticos para las muestras de luminancia y algunos de los elementos sintácticos para las muestras de crominancia se
40 codifican mediante CABAC y los otros elementos sintácticos para las muestras de luminancia y las muestras de crominancia se codifican por circunvalación. En algunos casos, los elementos sintácticos codificados mediante CABAC y codificados por circunvalación se entrelazan, dando como resultado un rendimiento de codificación de entropía no del todo óptimo debido a la información de control en las cadenas de procesamiento.

45 Por ejemplo, el modo de circunvalación tiene normalmente un rendimiento mucho mayor que el modo CABAC. Como un ejemplo, el modo de circunvalación puede procesar 6 *celdas* en un ciclo sin añadir más complejidad que el modo de circunvalación que procesa 1 *celda* por ciclo, mientras que el modo CABAC solo puede procesar 2 *celdas* por ciclo con un diseño mucho más complejo que el modo CABAC que
50 procesa 1 *celda* por ciclo. Esto se debe al cambio de contexto presente en el modo CABAC.

Para aumentar el rendimiento de la codificación de entropía, puede ser beneficioso agrupar entre sí elementos sintácticos codificados mediante CABAC y agrupar entre sí elementos sintácticos codificados por circunvalación. Como un ejemplo, los elementos sintácticos codificados mediante CABAC y los elementos
55 sintácticos codificados por circunvalación no se entrelazan entre sí. Como se describe en mayor detalle, las técnicas descritas en esta divulgación proporcionan varias técnicas de ejemplo para agrupar entre sí elementos sintácticos codificados mediante CABAC y agrupar entre sí elementos sintácticos codificados por circunvalación, lo que tiene la posibilidad de aumentar el rendimiento de la codificación de entropía.

60 Como un ejemplo, un codificador de vídeo codifica y señala en un flujo de bits codificado y un descodificador de vídeo recibe y descodifica a partir del flujo de bits codificado todas las celdas de intra-modo de luminancia y de intra-modo de crominancia codificadas mediante CABAC agrupadas entre sí y todas las celdas de intra-modo de luminancia y de intra-modo de crominancia codificadas por circunvalación agrupadas entre sí. Como otro ejemplo, un codificador de vídeo codifica y señala en un flujo de bits
65 codificado y un descodificador de vídeo recibe y descodifica a partir del flujo de bits codificado todas las celdas de intra-modo codificadas mediante CABAC para todos los bloques de una unidad de codificación y

5 todas las celdas de intra-modo codificadas por circunvalación agrupadas entre sí. Como otro ejemplo, un codificador de vídeo codifica y señala en un flujo de bits codificado y un descodificador de vídeo recibe y descodifica a partir del flujo de bits codificado todas las celdas de intra-modo de luminancia y crominancia codificadas mediante CABAC para todos los bloques de una unidad de codificación y todas las celdas de intra-modo de luminancia y crominancia codificadas por circunvalación agrupadas entre sí. En algunos casos se necesita más memoria para almacenar la sintaxis descodificada parcial para el descodificador de vídeo.

10 Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo para codificar y descodificar de manera más eficiente información de vídeo digital. Las técnicas de compresión de vídeo pueden definirse según una norma de codificación de vídeo. Los esfuerzos de la normalización HEVC se basan en un modelo de un dispositivo de codificación de vídeo denominado modelo de prueba HEVC (HM). El HM prevé mejoras en las capacidades de los dispositivos de codificación de vídeo actuales con respecto a los dispositivos de codificación de vídeo disponibles durante el desarrollo de otras normas de codificación de vídeo anteriores, por ejemplo la ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación de intra-predicción, la norma HEVC proporciona treinta y cinco modos de codificación de intra-predicción. Un borrador reciente (WD) de la norma HEVC, denominado "HEVC Working Draft 6" o "WD6", se describe en el documento JCTVC-H1003 de Bross et al., titulado "WD6: Working Draft 6 of High-Efficiency Video Coding (HEVC)" Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, octava conferencia: San José, California, EE.UU., febrero de 2011.

25 Además, otro borrador reciente de la norma HEVC, Borrador 8, se describe en el documento HCTVC-J1003_d7, de Bross, et al., titulado "High Efficiency Video Coding (HEVC) Text Specification Draft 8" Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, décima conferencia: Estocolmo, Suecia, julio de 2012. Un borrador más reciente de la norma HEVC, Borrador 9, está disponible, a fecha de 6 de marzo de 2013, en el sitio http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v10.zip. La norma HEVC también puede denominarse ISO/IEC 23008-HEVC, que es el número de la norma para la versión publicada de HEVC. Aunque las técnicas de esta divulgación se describen con respecto a la norma ITU-T H.264 y la norma HEVC, las técnicas de esta divulgación pueden aplicarse, por lo general, a cualquier norma de codificación de vídeo.

35 Una secuencia de vídeo incluye normalmente una serie de cuadros de vídeo, también denominados imágenes. Un grupo de imágenes (GOP) comprende generalmente una serie de uno o más cuadros de vídeo. Un GOP puede incluir datos sintácticos en la cabecera del GOP, en la cabecera de uno o más cuadros del GOP, o en otras ubicaciones, que indican el número de cuadros incluidos en el GOP. Cada cuadro puede incluir datos sintácticos de cuadro que describen el modo de codificación del cuadro respectivo. Cada cuadro de vídeo puede incluir una pluralidad de fragmentos. Cada fragmento puede incluir una pluralidad de bloques de vídeo. Un codificador de vídeo actúa normalmente en bloques de vídeo de cuadros de vídeo individuales con el fin de codificar los datos de vídeo. Los bloques de vídeo pueden tener tamaños fijos o variables y pueden tener un tamaño diferente según una norma de codificación específica.

45 Como un ejemplo, la norma ITU-T H.264 soporta intra-predicción para varios tamaños de bloque, tales como 16x16, 8x8 o 4x4 para componentes de luminancia, y 8x8 para componentes de crominancia, así como inter-predicción para varios tamaños de bloque, tales como 16x16, 16x8, 8x16, 8x8, 8x4, 4x8 y 4x4 para componentes de luminancia y tamaños escalados correspondientes para componentes de crominancia. En esta divulgación, "NxN" y "N por N" pueden usarse de manera intercambiable para hacer referencia a las dimensiones de píxel del bloque en lo que respecta a la dimensión vertical y la dimensión horizontal, por ejemplo 16x16 píxeles o 16 por 16 píxeles. En general, un bloque 16x16 tendrá 16 píxeles en la dirección vertical ($y = 16$) y 16 píxeles en la dirección horizontal ($x = 16$). Los píxeles también pueden denominarse muestras. Asimismo, un bloque NxN tiene generalmente N píxeles en la dirección vertical y N píxeles en la dirección horizontal, donde N representa un valor entero no negativo. Los píxeles de un bloque pueden estar ordenados en filas y columnas. Además, los bloques no necesitan tener necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender NxM píxeles, donde M no es necesariamente igual a N.

60 Los bloques de vídeo pueden comprender bloques de datos de píxel en el dominio de píxeles, o bloques de coeficientes de transformada en el dominio de transformada. Los coeficientes de transformada pueden generarse tras la aplicación de una transformada, tal como una transformada discreta del coseno (DCT), una transformada de número entero, una transformada de ondícula o una transformada conceptualmente similar, a los datos residuales de bloque de vídeo que representan diferencias de píxel entre bloques de vídeo codificados y bloques de vídeo predictivos. En algunos casos, un bloque de vídeo puede comprender bloques de coeficientes de transformada cuantificados en el dominio de transformada.

65 Bloques de vídeo más pequeños pueden proporcionar una mejor resolución y pueden usarse en

ubicaciones de un cuadro de vídeo que incluyen altos niveles de detalle. En general, los bloques y las diversas particiones, denominadas en ocasiones subbloques, pueden considerarse bloques de vídeo. Además, un fragmento puede considerarse una pluralidad de bloques de vídeo, tales como bloques y/o subbloques. Cada fragmento puede ser una unidad de un cuadro de vídeo que puede descodificarse de manera independiente. Como alternativa, los propios cuadros pueden ser unidades descodificables, u otras partes de un cuadro pueden definirse como unidades descodificables. El término "unidad codificada" puede referirse a cualquier unidad de un cuadro de vídeo que puede descodificarse de manera independiente, tal como un cuadro completo o un fragmento de un cuadro, a un grupo de imágenes (GOP) denominado también secuencia, o a otra unidad que pueda descodificarse de manera independiente definida según técnicas de codificación aplicables.

Tras una codificación intra-predictiva o inter-predictiva para generar datos predictivos y datos residuales, y tras cualquier transformada (tal como la transformada de número entero 4x4 u 8x8 usada en la norma H.264/AVC o una transformada discreta del coseno DCT) para generar coeficientes de transformada, puede llevarse a cabo la cuantificación de los coeficientes de transformada. La cuantificación se refiere generalmente a un proceso en el que los coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o todos los coeficientes. Por ejemplo, un valor de n bits puede redondearse por lo bajo a un valor de m bits durante la cuantificación, donde n es mayor que m .

Tras la cuantificación puede llevarse a cabo la codificación de entropía de los datos cuantificados, por ejemplo según la codificación de longitud variable adaptable al contenido (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación de entropía mediante división en intervalos de probabilidad (PIPE) u otra metodología de codificación de entropía. También puede llevarse a cabo la codificación de entropía de elementos sintácticos, tal como un elemento sintáctico que define un modo de predicción. Una unidad de procesamiento configurada para la codificación de entropía, u otra unidad de procesamiento, puede llevar a cabo otras funciones de procesamiento, tal como la codificación por longitud de series de cero de coeficientes cuantificados y/o la generación de información sintáctica, tal como valores de patrón de bloque codificado (CBP), el tipo de macrobloque, el modo de codificación, el tamaño máximo de macrobloque para una unidad codificada (tal como un cuadro, un fragmento, un macrobloque o una secuencia), o similar.

La norma HEVC se refiere a un bloque de datos de vídeo, tal como una unidad de codificación (CU). En general, las técnicas de esta divulgación se refieren a transformar, cuantificar, escanear y codificar por entropía datos de una CU. Una CU puede incluir una o más unidades de predicción (PU) y/o una o más unidades de transformada (TU). Las PU también pueden denominarse "particiones de predicción". Los datos sintácticos de un flujo de bits pueden definir la unidad de codificación más grande (LCU), que es la mayor unidad de codificación en lo que respecta al número de píxeles. En general, una LCU tiene una finalidad similar a la de un macrobloque de la norma H.264. Sin embargo, una LCU no está limitada necesariamente a un tamaño particular y puede incluir una o más CU. Una LCU incluye normalmente 64x64 píxeles. Las CU de una LCU incluyen normalmente 32x32, 16x16 u 8x8 píxeles. Por tanto, una LCU puede dividirse en varias sub-CU y cada sub-CU puede dividirse adicionalmente en varias sub-CU. Cada píxel de una CU puede incluir una componente de luminancia (Y), una componente de crominancia U (U) y una componente de crominancia V (V). En algunos ejemplos, las componentes de crominancia puede denominarse componentes C_r y C_b . En general, las referencias de esta divulgación a una CU pueden referirse a la unidad de codificación más grande (LCU) de una imagen o a una sub-CU de una LCU. Los datos sintácticos de un flujo de bits pueden definir el número máximo de veces que puede dividirse una LCU, lo que se denomina profundidad de CU. Por consiguiente, los datos sintácticos también pueden definir la unidad de codificación más pequeña (SCU). Esta divulgación también usa el término "bloque" para referirse a una CU, una PU o una TU.

Una LCU puede asociarse a una estructura de datos de árbol cuaternario. En general, una estructura de datos de árbol cuaternario incluye un nodo por CU, donde el nodo raíz corresponde a la LCU. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hoja, cada uno de los cuales corresponde a una de las sub-CU. Cada nodo de la estructura de datos de árbol cuaternario puede proporcionar datos sintácticos para la CU correspondiente. Por ejemplo, un nodo del árbol cuaternario puede incluir un indicador de división, que indica si la CU correspondiente al nodo está dividida en varias sub-CU. Los elementos sintácticos de una CU pueden definirse de manera recursiva y pueden depender de si la CU está dividida en varias sub-CU. Si una CU no está dividida adicionalmente, se denomina CU hoja. En esta divulgación, cuatro sub-CU de una CU hoja también se denominarán CU hoja aunque no haya una división explícita de la CU hoja original. Por ejemplo, si una CU con un tamaño de 16x16 no se divide adicionalmente, las cuatro sub-CU 8x8 también se denominarán CU hoja aunque la CU 16x16 no se haya dividido.

Además, las TU de las CU hoja también pueden asociarse a estructuras de datos respectivas de árbol cuaternario. Es decir, una CU hoja puede incluir un árbol cuaternario que indica cómo la CU hoja está

dividida en varias TU. Esta divulgación hace referencia al árbol cuaternario que indica cómo una LCU está dividida como un árbol cuaternario de CU, indicando el árbol cuaternario cómo una CU hoja está dividida en varias TU como un árbol cuaternario de TU. El nodo raíz de un árbol cuaternario de TU corresponde generalmente a una CU hoja, mientras que el nodo raíz de un árbol cuaternario de CU corresponde generalmente a una LCU. Las TU del árbol cuaternario de TU que no están divididas se denominan TU hoja.

Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa la totalidad o una parte de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Por ejemplo, cuando la PU está codificada en el modo de inter-predicción, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, una precisión de un cuarto de píxel o una precisión de un octavo de píxel), una dirección de predicción que identifica una lista de imágenes de referencia (por ejemplo, una lista 0 o una lista 1) para el vector de movimiento y/o un valor de índice de imagen de referencia que indica un cuadro de referencia de la lista de imágenes de referencia al que apunta el vector de movimiento. De manera similar, cuando una CU está codificada en el modo de intra-predicción, una PU puede incluir datos que definen un modo de intra-predicción (por ejemplo, una predicción angular, una predicción DC, una predicción plana, etc.).

Los datos de la CU hoja que definen las PU también pueden describir, por ejemplo, una división de la CU en una o más PU. Una división de una CU en una o más PU puede denominarse modo de división. Los modos de división que están disponibles para una CU puede diferir en función de si la CU está sin codificar, del modo de intra-predicción codificado o del modo de inter-predicción codificado. En la intra-codificación, una PU puede tratarse de mismo modo que una unidad de transformada de hoja descrita a continuación.

Una CU hoja puede incluir una o más unidades de transformada (TU). En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a no ser que se indique lo contrario. Las unidades de transformada pueden especificarse usando una estructura de árbol cuaternario de TU, como la descrita anteriormente. Es decir, un indicador de división puede indicar si una CU hoja está dividida en cuatro unidades de transformada. Después, cada unidad de transformada puede dividirse adicionalmente en cuatro sub-TU. Cuando una TU no está dividida adicionalmente, puede denominarse TU hoja. En general, un indicador de división puede indicar que una TU hoja está dividida en varias TU de forma cuadrada. Con el fin de indicar que una TU está dividida en varias TU que no tienen una forma cuadrada, pueden incluirse otros datos sintácticos, por ejemplo datos sintácticos que indican que las TU van a dividirse según la NSQT.

Generalmente, en lo que respecta a la intra-codificación, todas las TU hoja que pertenecen a una CU hoja pueden compartir el mismo modo de intra-predicción. Es decir, el mismo modo de intra-predicción puede aplicarse para calcular valores predichos para todas las TU de una CU hoja. En lo que respecta a la intra-codificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando el modo de intra-predicción, como una diferencia entre la parte de los valores predictivos correspondientes a la TU y el bloque original. El valor residual puede transformarse, cuantificarse y examinarse, como se ha descrito anteriormente. En lo que respecta a la inter-codificación, un codificador de vídeo puede llevar a cabo la predicción en el nivel de PU y puede calcular un valor residual para cada PU. Los valores residuales correspondientes a una CU hoja pueden transformarse, cuantificarse y examinarse. En lo que respecta a la inter-codificación, una TU hoja puede ser mayor o menor que una PU. En lo que respecta a la intra-codificación, una PU puede estar coubicada con una TU hoja correspondiente. En algunos ejemplos, el tamaño máximo de una TU hoja puede ser el tamaño de la CU hoja correspondiente.

En la norma HEVC, el número de modos de intra-predicción usados para generar un bloque predictivo para una CU se determina mediante la estructura de PU. Una estructura de PU INTRA_2Nx2N indica que una CU solo incluye una PU. Una estructura de PU INTRA_NxN indica que una CU incluye cuatro PU. Además, cada PU de una CU puede tener un modo de intra-predicción de luminancia y las PU de una CU pueden compartir un modo de predicción de crominancia. Por ejemplo, una CU con una estructura de PU INTRA_2Nx2N puede tener un modo de intra-predicción de luminancia y un modo de intra-predicción de crominancia. Además, una CU con una estructura de PU INTRA_NxN puede tener cuatro modos de intra-predicción de luminancia (es decir, uno por cada PU) y un modo de intra-predicción de crominancia para toda la CU (es decir, las cuatro PU comparten un modo de predicción de crominancia). Los elementos sintácticos pueden generarse para señalar modos de intra-predicción.

La FIG. 1 es un diagrama conceptual que ilustra ejemplos de estructuras de unidades de predicción y modos de codificación de intra-predicción asociados. Como se ilustra en la FIG. 1, la estructura de unidad de predicción (PU) de INTRA_2Nx2N incluye solamente una PU y la estructura de PU INTRA_NxN incluye cuatro PU. Como se ilustra en la FIG. 1, la estructura de PU INTRA_NxN es simétrica e incluye cuatro PU cuadradas de igual tamaño. Como se ha descrito anteriormente, cada una de las cuatro PU ilustradas en la

FIG. 1 puede tener una intra-predicción de luminancia y compartir una intra-predicción de crominancia. Debe observarse que la norma HEVC define otras estructuras de PU para su uso con modos de inter-predicción. Algunas de estas PU no son simétricas y/o incluyen PU rectangulares. Aunque las técnicas de esta divulgación se describen con respecto a las estructuras de PU INTRA_2Nx2N y de PU INTRA_NxN, las técnicas pueden aplicarse generalmente a estructuras de PU adicionales que pueden usarse con modos de intra-predicción. Por ejemplo, las técnicas de esta divulgación pueden aplicarse en caso de que una estructura de PU de cuatro PU rectangulares esté asociada a un modo de codificación de intra-predicción.

Como se ha descrito anteriormente, cuando una CU se codifica usando un modo de intra-predicción, pueden usarse elementos sintácticos para señalar los modos de intra-predicción. En algunos ejemplos se usan tres elementos sintácticos para señalar un modo de intra-predicción de luminancia para una PU y se usa un elemento sintáctico para señalar un modo de intra-predicción de crominancia. En la Tabla 1 se muestran en negrita los elementos sintácticos.

Tabla 1

prev_intra_luma_pred_flag [x0] [y0]
si(prev_intra_luma_pred_flag [x0] [y0])
mpm_idx [x0] [y0]
Si no
rem_intra_luma_pred_mode [x0] [y0]
intra_chroma_pred_mode [x0] [y0]

Los elementos sintácticos **prev_intra_luma_pred_flag** [x0] [y0], **mpm_idx**[x0] [y0] y **rem_intra_luma_pred_mode**[x0] [y0] especifican el modo de intra-predicción para las muestras de luminancia. Los índices de matriz x0, y0 especifican la ubicación de la muestra de luminancia superior izquierda de un bloque de predicción con respecto a la muestra de luminancia superior izquierda de la imagen. El elemento sintáctico **prev_intra_luma_pred_flag**[x0] [y0] indica una condición verdadera o falsa y puede tener un valor de 1 o 0. En un ejemplo, cuando **prev_intra_luma_pred_flag**[x0] [y0] es igual a 1, el modo de intra-predicción para una PU actual se infiere a partir de una PU intra-predicha vecina.

Por ejemplo, como se indica en la Tabla 1, si **prev_intra_luma_pred_flag**[x0] [y0] es igual a 1, un codificador de vídeo señala en el flujo de bits codificado y un descodificador de vídeo recibe el elemento sintáctico **mpm_idx**[x0] [y0] en el flujo de bits codificado. El elemento sintáctico **mpm_idx**[x0] [y0] es un índice hacia una lista de modos más probables (mpm). Por ejemplo, el codificador de vídeo y el descodificador de vídeo pueden estar configurados para generar la lista de modos más probables. La lista de modos más probables identifica los modos de intra-predicción.

El descodificador de vídeo puede generar de manera implícita la lista de modos más probables (es decir, sin recibir señalización del codificador de vídeo que indica qué modos de intra-predicción van a incluirse en la lista de modos más probables) o puede generar la lista de modos más probables en función de señalización recibida desde el codificador de vídeo. En cualquiera de los ejemplos, la lista de modos más probables en el lado del codificador de vídeo y la lista de modos más probables en el lado del descodificador de vídeo pueden identificar los mismos modos de intra-predicción y en el mismo orden. En algunos ejemplos, el codificador de vídeo y el descodificador de vídeo pueden generar la lista de modos más probables para cada PU intra-predicha de la CU, y la lista de modos más probables para dos o más PU de la CU pueden ser diferentes (es decir, puede haber listas respectivas de modos más probables para las PU de la CU). Sin embargo, las técnicas descritas en esta divulgación no están limitadas a esto.

Puede haber varias maneras en las que el codificador de vídeo y el descodificador de vídeo pueden generar la lista de modos más probables. Como un ejemplo, el codificador de vídeo y el descodificador de vídeo están configurados para identificar un número fijo de modos de intra-predicción en la lista de modos más probables (es decir, identificar tres modos de intra-predicción). El codificador de vídeo y el descodificador de vídeo evalúan el modo de intra-predicción de una primera PU vecina que está cerca de la PU actual (por ejemplo, la PU que está intra-prediciéndose) y una segunda PU vecina que está cerca de la PU actual. Ejemplos de la primera y la segunda PU vecinas incluyen la PU izquierda, la PU superior izquierda, la PU derecha, la PU inferior y, en general, cualquier PU vecina.

El codificador de vídeo y el descodificador de vídeo incluyen los modos de intra-predicción de la primera y la segunda PU vecinas de la lista de modos más probables. Si una o ambas de la primera y la segunda PU vecinas no están intra-predichas, el codificador de vídeo y el descodificador de vídeo pueden identificar el modo de intra-predicción DC, por ejemplo, en la lista de modos más probables en las ubicaciones correspondientes de la lista de modos más probables para la primera y la segunda PU vecinas.

Para determinar el tercer modo de intra-predicción en la lista de modos de intra-predicción, el codificador de vídeo y el descodificador de vídeo pueden determinar si los modos de intra-predicción para la primera y la segunda PU vecinas son iguales (incluyendo si el modo de intra-predicción DC se seleccionó en caso de que no se intra-predijera). Si los modos de intra-predicción para la primera y la segunda PU vecinas son iguales, el codificador de vídeo y el descodificador de vídeo pueden implementar una primera técnica para identificar el tercer modo de intra-predicción en la lista de modos de intra-predicción. En algunos ejemplos, si los modos de intra-predicción para la primera y la segunda PU vecinas son iguales, el codificador de vídeo y el descodificador de vídeo pueden identificar uno de los modos de intra-predicción de la lista de modos más probables e identificar el segundo y el tercer modo de intra-predicción de la lista de modos más probables usando la primera técnica (u otra técnica, en lugar de la primera técnica). Si los modos de intra-predicción para la primera y la segunda PU vecinas son diferentes, el codificador de vídeo y el descodificador de vídeo pueden implementar una segunda técnica para identificar el tercer modo de intra-predicción.

Puede haber varios ejemplos de la primera y la segunda técnica para identificar el tercer modo de intra-predicción, y las técnicas descritas en esta divulgación no están limitadas a una técnica particular. En algunos ejemplos, el tercer modo de intra-predicción basado en la primera o en la segunda técnica no necesita ser necesariamente un modo de intra-predicción de una PU vecina. En algunos ejemplos, el tercer modo de intra-predicción basado en la primera o en la segunda técnica puede basarse en un modo de intra-predicción de una PU vecina.

El elemento sintáctico **mpm_idx**[x0][y0] es el índice hacia la lista de modos más probables que el codificador de vídeo señala y el descodificador de vídeo recibe. En este ejemplo, a partir del valor de **mpm_idx**[x0][y0] el descodificador de vídeo puede configurarse para determinar el modo de intra-predicción para la PU actual. En algunos ejemplos, puesto que la lista de modos más probables está basada en los modos de intra-predicción de PU vecinas (por ejemplo, la primera y la segunda PU vecina), el elemento sintáctico **mpm_idx**[x0][y0] puede usarse, en algunos casos, para identificar la PU vecina específica a usar para inferir el modo de intra-predicción para la PU actual. Por ejemplo, si **mpm_idx**[x0][y0] es un índice hacia la lista de modos más probables que identifica el modo de intra-predicción asociado a la primera PU vecina, puede considerarse que el descodificador de vídeo identifica la primera PU vecina como la PU a partir de la cual va a inferirse el modo de intra-predicción de la PU actual.

Por ejemplo, haciendo referencia a la FIG. 1, el modo de intra-predicción de PU₁ puede ser igual al modo de intra-predicción de PU₀ en el ejemplo de INTRA_NxN. El elemento sintáctico **mpm_idx**[x0][y0] puede indicar a partir de qué PU vecina de un conjunto de PU intra-predichas vecinas, una PU actual puede inferir un modo de intra-predicción. En este ejemplo, si **prev_intra_luma_pred_flag**[x0][y0] es igual a 0, el modo de intra-predicción no se infiere a partir de una PU vecina, sino que se indica por el elemento sintáctico **rem_intra_luma_pred_mode**[x0][y0].

El elemento sintáctico **rem_intra_luma_pred_mode**[x0][y0] puede especificar el modo de intra-predicción particular para la PU actual. En algunos ejemplos, el elemento sintáctico **rem_intra_luma_pred_mode**[x0][y0] puede especificar un modo de intra-predicción para la PU actual que no está incluido en los modos de intra-predicción identificados por la lista de modos más probables. Por ejemplo, el elemento sintáctico **rem_intra_luma_pred_mode**[x0][y0] identifica un modo de intra-predicción restante a partir de todos los modos de intra-predicción posibles cuando los modos de intra-predicción identificados en la lista de modos más probables están excluidos. Sin embargo, las técnicas de esta divulgación no están limitadas a esto.

El elemento sintáctico **intra_chroma_pred_mode**[x0][y0] especifica el modo de intra-predicción para las muestras de crominancia de una CU. Una intra-predicción de crominancia puede basarse en un modo de intra-predicción de luminancia. Un elemento sintáctico **chroma_pred_from_luma_enabled_flag** puede indicar si el modo de intra-predicción para la crominancia está basado en el modo de intra-predicción para la luminancia.

En el borrador WD6 de la norma HEVC, los posibles modos de intra-predicción incluyen un modo de predicción plana (predMode = 0), una predicción DC (predMode = 1), 33 modos de predicción angular (predMode = 2,..., 34) y un modo de predicción para una componente de crominancia que infiere un modo de intra-predicción a partir de un modo de predicción de luminancia (predMode = 35). Por tanto, para una componente de luminancia, dependiendo del valor de **prev_intra_luma_pred_flag**, el elemento sintáctico **mpm_idx** puede indicar un índice hacia la lista de modos más probables, donde cada entrada de la lista de modos más probables identifica uno de 35 modos de intra-predicción posibles, o el elemento sintáctico **rem_intra_luma_pred_mode** puede indicar uno de 35 modos de intra-predicción posibles (excluyendo esos modos de intra-predicción identificados en la lista de modos más probables, en algunos ejemplos). En un ejemplo, el elemento sintáctico **rem_intra_luma_pred_mode** puede tener un valor entero comprendido entre 0 y 31 y puede codificarse usando un procedimiento de codificación binaria de longitud fija, y el elemento sintáctico **mpm_idx** puede tener un valor entero comprendido entre 0 y 2 (por ejemplo, para

identificar una de las tres entradas de la lista de modos más probables) y puede codificarse usando un procedimiento de codificación unaria truncada. Además, para las componentes de crominancia, **intra_chroma_pred_mode** y uno o más elementos sintácticos pueden indicar uno de 36 modos de intra-predicción posibles. En un ejemplo, el elemento sintáctico **intra_chroma_pred_mode** puede tener un valor entero comprendido entre 0 y 5. Una descripción más detallada de cómo cada uno de los elementos sintácticos **mpm_idx**, **rem_intra_luma_pred_mode** e **intra_chroma_pred_mode** identifican modos de intra-predicción particulares se proporciona en el borrador WD6 de la norma HEVC y, en aras de la brevedad, se repite en el presente documento. Sin embargo, debe observarse que las técnicas descritas en el presente documento pueden aplicarse generalmente a variaciones contempladas de los elementos sintácticos **mpm_idx**, **rem_intra_luma_pred_mode** e **intra_chroma_pred_mode**.

Como se ha descrito anteriormente, una norma de codificación de vídeo puede codificar por entropía elementos sintácticos según una técnica CABAC. Para aplicar la codificación CABAC a un elemento sintáctico, un codificador de vídeo puede llevar a cabo la binarización en un elemento sintáctico. La binarización se refiere al proceso de convertir un valor sintáctico en una serie de uno o más bits. Estos bits pueden denominarse "*celdas*". La binarización es un proceso sin pérdidas y puede incluir una sola o una combinación de las siguientes técnicas de codificación: codificación de longitud fija, codificación unaria, codificación unaria truncada, codificación de Rice truncada, codificación de Golomb, codificación de Golomb exponencial y codificación de Golomb-Rice. Por ejemplo, la binarización puede incluir representar el valor entero de 5 como 0000101 usando una técnica de longitud fija de 8 bits o como 11110 usando una técnica de codificación unaria.

Después de la binarización, un codificador de vídeo puede identificar un contexto de codificación. El contexto de codificación puede identificar probabilidades de codificación de *celdas* que tienen valores particulares. Por ejemplo, un contexto de codificación puede indicar una probabilidad de 0,7 de codificar una celda de valor 0 y una probabilidad de 0,3 de codificar una celda de valor 1. Después de identificar el contexto de codificación, el codificador de vídeo puede codificar de manera aritmética esa celda basándose en el contexto. Las celdas codificadas usando una codificación CABAC convencional pueden denominarse "*celdas CABAC*".

Además, en lugar de llevar a cabo una codificación CABAC convencional en todas las celdas de elementos sintácticos, un codificador de vídeo puede codificar algunas *celdas* de elementos sintácticos usando codificación CABAC por circunvalación. La codificación por circunvalación se refiere al proceso de codificar de manera aritmética una celda sin usar un contexto adaptativo. Es decir, el motor de codificación por circunvalación no selecciona contextos y puede suponer una probabilidad de 0,5 para ambos símbolos (0 y 1). Aunque la codificación por circunvalación puede no ser tan eficiente en lo que respecta al ancho de banda como la codificación CABAC, llevar a cabo la codificación por circunvalación en una celda puede requerir menos esfuerzo computacional que llevar a cabo una codificación CABAC convencional en la celda. Además, llevar a cabo la codificación por circunvalación puede permitir un mayor grado de paralelización y rendimiento. Las celdas codificadas usando la codificación por circunvalación pueden denominarse "*celdas de circunvalación*".

Cuando se implementa un codificador CABAC, puede incluir un motor de codificación convencional para llevar a cabo una codificación CABAC convencional y un motor de codificación por circunvalación para llevar a cabo una codificación por circunvalación. Si una celda se codifica mediante CABAC, se usa el motor de codificación CABAC convencional para codificar esta celda. El motor de codificación CABAC convencional puede necesitar más de dos ciclos de procesamiento para codificar una única *celda*. Sin embargo, con un diseño adecuado de cadenas de procesamiento, un motor de codificación CABAC convencional puede necesitar solamente $n+M$ ciclos para codificar n *celdas*, donde M es la información de control para iniciar la cadena de procesamiento. Por lo general, M es mayor que 0.

Al inicio del proceso de codificación CABAC (es decir, en cada cambio del modo por circunvalación al modo convencional), se introduce información de control de cadena de procesamiento. Si una celda se codifica por circunvalación, se usa el motor de codificación por circunvalación para codificar esta *celda*. Cabe esperar que el motor de codificación por circunvalación necesite solamente un ciclo para codificar un elemento sintáctico de n bits, donde n puede ser mayor que uno. Por tanto, el número total de ciclos para codificar un conjunto de *celdas* de circunvalación y de *celdas* CABAC puede reducirse si todas las *celdas* de circunvalación del conjunto se codifican juntas y todas las *celdas* CABAC del conjunto se codifican juntas. En particular, codificar juntas las celdas de circunvalación antes o después de pasar a la codificación CABAC puede ahorrar la información de control requerida para volver a activar el motor de codificación CABAC.

Como se ha descrito anteriormente, una CU con una estructura de PU INTRA_NxN puede tener cuatro modos de intra-predicción de luminancia y, por tanto, cuatro conjuntos de elementos sintácticos **prev_intra_luma_pred_flag**, **mpm_idx** y **rem_intra_luma_pred_mode**. Entre estos tres elementos sintácticos, en un ejemplo, solo **prev_intra_luma_pred_flag** se codifica mediante CABAC y tanto **mpm_idx**

como **rem_intra_luma_pred_mode** se codifican por circunvalación. De nuevo, la codificación por circunvalación puede referirse a codificar *celdas* con un proceso diferente a CABAC, es decir, sin usar el motor CABAC. Ejemplos de codificación diferente a CABAC que pueden usarse en la codificación por circunvalación incluyen la codificación de Golomb, la codificación de Golomb exponencial y la codificación de Golomb-Rice.

Como se ha descrito anteriormente, una CU con una estructura de PU INTRA_NxN puede tener un modo de intra-predicción de crominancia que se usa para las cuatro PU. El modo de intra-predicción de crominancia puede basarse en los modos de intra-predicción de luminancia. La Tabla 2 muestra la binarización del elemento sintáctico **intra_chroma_pred_mode** como un ejemplo. La binarización da como resultado uno o más *celdas* para representar el valor correspondiente de **intra_chroma_pred_mode**. En este ejemplo, las celdas que están subrayados se han codificado mediante CABAC y el resto de las celdas se han codificado por circunvalación. Como se ilustra en la Tabla 2, la binarización de **intra_chroma_pred_mode** y si una celda particular se ha codificado mediante CABAC o se ha codificado por circunvalación se basan en el valor de **chroma_pred_from_luma_enable_flag**. Por tanto, la binarización de un elemento sintáctico y si una celda particular se ha codificado mediante CABAC o se ha codificado por circunvalación pueden basarse en el valor de otro elemento sintáctico (por ejemplo, **chroma_pred_from_luma_enable_flag**).

Tabla 2 Binarización de **intra_chroma_pred_mode** en función de **chroma_pred_from_luma_enabled_flag**

Valor de intra_chroma_pred_mode	chroma_pred_from_luma_enabled_flag = 1	chroma_pred_from_luma_enabled_flag = 0
5	<u>0</u>	n.a.
4	<u>10</u>	<u>0</u>
0	<u>1100</u>	<u>100</u>
1	<u>1101</u>	<u>101</u>
2	<u>1110</u>	<u>110</u>
3	<u>1111</u>	<u>111</u>

Las siguientes Tablas 3 y 4 representan las estructuras de codificación de elementos sintácticos que identifican los modos de predicción de luminancia y crominancia para una CU en algunos ejemplos. La Tabla 3 ilustra la codificación de elementos sintácticos para una estructura de PU INTRA_2Nx2N y la Tabla 4 ilustra la codificación de una estructura de PU INTRA_NxN. En las Tablas 3 y 4, el elemento sintáctico **intra_chroma_pred_mode** se representa como los elementos sintácticos **chroma_mode_cabac_bins** y **chroma_mode_bypass_bins**, donde **chroma_mode_cabac_bins** denota las celdas que están codificadas mediante CABAC y **chroma_mode_bypass_bins** denota las celdas que están codificadas por circunvalación, como se proporciona en la Tabla 2. Las Tablas 3 y 4 identifican si los elementos sintácticos se han codificado usando un modo CABAC o un modo de circunvalación. Los elementos sintácticos se proporcionan a un motor de codificación CABAC y normalmente se codifican por entropía en el orden en que se presentan en las Tablas 3 y 4. Además, como se ilustra en la Tabla 4, hay cuatro conjuntos de **prev_intra_luma_pred_flag**, **mpm_idx** y **rem_intra_luma_pred_mode**, cada uno correspondiendo a una de cuatro PU, y una indicación de modo de crominancia compartida por las cuatro PUs.

Tabla 3 Estructura de codificación intra-modo para **intra_2Nx2N**

prev_intra_luma_pred_flag[x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

Tabla 4 Estructura de codificación intra-modo para **intra_NxN**

prev_intra_luma_pred_flag[x0][y0]	(CODIFICADO MEDIANTE CABAC)
----------------------------------------------	------------------------------------

si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
//CODIFICACIÓN DE LUMINANCIA PARA PRIMERA PU	
prev_intra_luma_pred_flag[x1][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y0])	
mpm_idx[x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
//CODIFICACIÓN DE LUMINANCIA PARA SEGUNDA PU	
prev_intra_luma_pred_flag[x0][y1]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y1])	
mpm_idx[x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
//CODIFICACIÓN DE LUMINANCIA PARA TERCERA PU	
prev_intra_luma_pred_flag[x1][y1]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y1])	
mpm_idx[x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
//CODIFICACIÓN DE LUMINANCIA PARA CUARTA PU	
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
//CODIFICACIÓN DE CROMINANCIA PARA TODAS LAS PUs	

5 Como se ilustra en las Tablas 3 y 4, los elementos sintácticos que están codificados mediante CABAC están entrelazados con elementos sintácticos que están codificados por circunvalación, lo que, como se ha descrito anteriormente, puede aumentar el número total de ciclos requeridos para codificar por entropía todos los elementos sintácticos de las Tablas 3 y 4, por ejemplo debido a la información de configuración para volver a activar el motor de codificación CABAC. Como se ilustra en la Tabla 4, el problema se agrava si una CU contiene múltiples PU. Con el fin de reducir la información de control total en las cadenas de procesamiento y el número total de ciclos requeridos para codificar los elementos sintácticos que identifican los modos de intra-predicción, pueden realizarse ciertos ajustes en el orden de codificación según esta divulgación.

10 La FIG. 2 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo 10 de ejemplo que puede utilizar técnicas según esta divulgación. Como se muestra en la FIG. 2, el sistema 10 incluye un dispositivo origen 12 que señala datos de vídeo codificados. En algunos ejemplos, el dispositivo origen 12 puede señalar los datos de vídeo codificados al dispositivo destino 14 a través de un canal de comunicación 16. El dispositivo origen 12 y el dispositivo destino 14 pueden comprender cualquiera de una amplia gama de dispositivos. En algunos casos, el dispositivo origen 12 y el dispositivo destino 14 pueden comprender dispositivos de comunicación inalámbricos, tales como microteléfonos inalámbricos, los denominados radioteléfonos celulares o vía satélite, o cualquier dispositivo inalámbrico que pueda comunicar información de vídeo a través de un canal de comunicación 16, donde el canal de comunicación

16 es inalámbrico en este caso. Sin embargo, las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Por ejemplo, estas técnicas pueden aplicarse a las radiodifusiones televisivas inalámbricas, a las transmisiones televisivas por cable, a las transmisiones televisivas vía satélite, a las transmisión de vídeo por Internet, al vídeo digital codificado que está codificado en un medio de almacenamiento y a otros escenarios. Por consiguiente, el canal de comunicación 16 puede comprender cualquier combinación de medios inalámbricos, cableados o de almacenamiento adecuados para la transmisión o el almacenamiento de datos de vídeo codificados para su posterior recuperación por el dispositivo destino 14.

En el ejemplo de la FIG. 2, el dispositivo origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20, un modulador / desmodulador (módem) 22 y un transmisor 24. El dispositivo destino 14 incluye un receptor 26, un módem 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. Según esta divulgación, el codificador de vídeo 20 del dispositivo origen 12 puede estar configurado para aplicar las técnicas de codificación de entropía de tal manera que se reduzca la información de control en las cadenas de procesamiento. En otros ejemplos, un dispositivo origen y un dispositivo destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo origen 12 puede recibir datos de vídeo desde una fuente de vídeo externa 18, tal como una cámara externa. Asimismo, el dispositivo destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

El sistema ilustrado 10 de la FIG. 2 es simplemente un ejemplo. Las técnicas para codificar por entropía un conjunto de indicadores y un elemento sintáctico respectivo para cada indicador de manera que se reduzca la información de control en las cadenas de procesamiento pueden llevarse a cabo por cualquier dispositivo de codificación y/o decodificación de vídeo digital. Aunque, por lo general, las técnicas de esta divulgación se llevan a cabo por un dispositivo de codificación de vídeo, las técnicas también pueden llevarse a cabo mediante un codificador / decodificador de vídeo, denominado habitualmente "códec". Además, las técnicas de esta divulgación también pueden llevarse a cabo por un preprocesador de vídeo. El dispositivo origen 12 y el dispositivo destino 14 son simplemente ejemplos de tales dispositivos de codificación, donde el dispositivo origen 12 genera datos de vídeo codificados para su transmisión al dispositivo destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de manera casi simétrica, de modo que cada uno de los dispositivos 12, 14 incluye componentes de codificación y de decodificación de vídeo. Por tanto, el sistema 10 puede soportar una transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo para el flujo continuo de vídeo, la reproducción de vídeo, la radiodifusión de vídeo o la videotelefonía.

La fuente de vídeo 18 del dispositivo origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo grabado previamente y/o vídeo procedente de un proveedor de contenido de vídeo. Como una alternativa adicional, la fuente de vídeo 18 puede generar vídeo fuente como datos basados en gráficos de ordenador, o una combinación de vídeo en directo, vídeo de un archivo y vídeo generado por ordenador. En algunos casos, si la fuente de vídeo 18 es una videocámara, el dispositivo origen 12 y el dispositivo destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, como se ha mencionado anteriormente, las técnicas descritas en esta divulgación pueden aplicarse a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el vídeo grabado, pregrabado o generado por ordenador puede codificarse por el codificador de vídeo 20. La información de vídeo codificado puede modularse después por el módem 22, según una norma de comunicación, y transmitirse al dispositivo destino 14 a través del transmisor 24. El módem 22 puede incluir varios mezcladores, filtros, amplificadores u otros componentes diseñados para la modulación de señales. El transmisor 24 puede incluir circuitos diseñados para transmitir datos, incluyendo amplificadores, filtros y una o más antenas.

El receptor 26 del dispositivo destino 14 recibe información a través del canal 16, y el módem 28 desmodula la información. De nuevo, el proceso de codificación de vídeo puede implementar una o más de las técnicas descritas en el presente documento para codificar por entropía elementos sintácticos con el fin de reducir la información de control en las cadenas de procesamiento. La información comunicada a través del canal 16 puede incluir información sintáctica definida por el codificador de vídeo 20, que también es usada por el decodificador de vídeo 30, que incluye elementos sintácticos que describen características y/o el procesamiento de macrobloques u otras unidades codificadas, por ejemplo grupos de imágenes (GOP). El dispositivo de visualización 32 muestra los datos de vídeo descodificados a un usuario y puede comprender cualquiera de una variedad de dispositivos de visualización, tal como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos de emisión de luz (OLED) u otro tipo de dispositivo de visualización.

En el ejemplo de la FIG. 2, el canal de comunicación 16 puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física, o cualquier combinación de medios inalámbricos y cableados. El canal de comunicación 16 puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área

extensa o una red global tal como Internet. El canal de comunicación 16 representa generalmente cualquier medio de comunicación adecuado, o un conjunto de diferentes medios de comunicación, para transmitir datos de vídeo desde el dispositivo origen 12 hasta el dispositivo destino 14, incluyendo cualquier combinación adecuada de medios cableados o inalámbricos. El canal de comunicación 16 puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo origen 12 al dispositivo destino 14. En otros ejemplos, el dispositivo origen 12 puede almacenar datos codificados en un medio de almacenamiento, en lugar de transmitir los datos. Asimismo, el dispositivo destino 14 puede estar configurado para recuperar datos codificados de un medio de almacenamiento. De esta manera, el dispositivo origen 12 puede estar configurado para generar un producto de programa informático, donde el producto de programa informático incluye un archivo de vídeo codificado según las técnicas descritas en el presente documento.

El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar según una norma de compresión de vídeo, tal como las descritas en el presente documento. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Aunque no se muestra en la FIG. 2, en algunos aspectos el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar integrados en un codificador y un descodificador de audio, y pueden incluir unidades de multiplexación y desmultiplexación apropiadas, u otro hardware y software, para llevar a cabo la codificación de audio y vídeo en un flujo de datos común o en flujos de datos diferentes. Si procede, las unidades de multiplexación y desmultiplexación pueden ajustarse al protocolo de multiplexación ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

El codificador de vídeo 20 y el descodificador de vídeo 30 pueden implementarse como cualquiera de una variedad de sistemas de circuitos de codificación o descodificación adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas de campo programable (FPGA), dispositivos de comunicación inalámbricos que incluyen un dispositivo de codificación de vídeo, tal como un codificador o un descodificador, lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden incluirse en uno o más codificadores o descodificadores, donde cualquiera de los mismos puede estar integrado como parte de un codificador / descodificador combinado (códec) en una cámara respectiva, ordenador, dispositivo móvil, dispositivo de abonado, dispositivo de radiodifusión, codificador, servidor o similar. Un aparato que incluye un codificador de vídeo 20 y/o un descodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrico, tal como un teléfono celular.

El codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar configurados para implementar técnicas según esta divulgación. Tales técnicas pueden reducir la información de control en las cadenas de procesamiento, aumentando posiblemente de este modo el rendimiento. Por ejemplo, una causa de la sobrecarga en las cadenas de procesamiento es la información de control asociada a los ciclos, necesaria para iniciar la codificación aritmética binaria adaptable al contexto (CABAC). En las técnicas descritas en esta divulgación, el codificador de vídeo 20 codifica por entropía un primer grupo de elementos sintácticos usando codificación CABAC (por ejemplo, con un motor CABAC convencional), y codifica por entropía un segundo grupo de elementos sintácticos usando codificación por circunvalación (por ejemplo, con un motor de circunvalación). Asimismo, el descodificador de vídeo 30 descodifica por entropía el primer grupo de elementos sintácticos usando descodificación CABAC (por ejemplo, con un motor CABAC convencional), y descodifica por entropía el segundo grupo de elementos sintácticos usando descodificación por circunvalación (por ejemplo, con un motor de descodificación por circunvalación).

De esta manera, los elementos sintácticos del primer grupo que están codificados mediante CABAC no se entrelazan con elementos sintácticos del segundo grupo que están codificados por circunvalación en el flujo de bits. Si los elementos sintácticos codificados mediante CABAC y los elementos sintácticos codificados por circunvalación se entrelazaran, entonces el codificador de vídeo 20 y el descodificador de vídeo 30 tendrían que reactivar repetidamente el motor CABAC.

Por ejemplo, supóngase que un primer elemento sintáctico va a codificarse mediante CABAC y que precede a un segundo elemento sintáctico que va a codificarse por circunvalación, y que un tercer elemento sintáctico que va a codificarse mediante CABAC sigue al segundo elemento sintáctico que va a codificarse por circunvalación. En este ejemplo, en el que los elementos sintácticos que van a codificarse mediante CABAC y por circunvalación están entrelazados, el codificador de vídeo 20 tendría que activar el motor CABAC para el primer elemento sintáctico y después volver a activar el motor CABAC para el tercer elemento sintáctico después de codificar el segundo elemento sintáctico con el motor de circunvalación, lo que malgasta ciclos.

En las técnicas descritas en esta divulgación, codificando mediante CABAC el primer grupo de elementos sintácticos y codificando por circunvalación el segundo grupo de elementos sintácticos, el codificador de vídeo 20 puede reducir el número de veces que tiene que activarse el motor CABAC, reduciendo así el

número de ciclos malgastados. Asimismo, descodificando mediante CABAC el primer grupo de elementos sintácticos y descodificando por circunvalación el segundo grupo de elementos sintácticos, el descodificador de vídeo 30 puede reducir el número de veces que tiene que activarse el motor CABAC, reduciendo así el número de ciclos malgastados.

5

Como se describe en mayor detalle, cada elemento sintáctico del primer grupo de elementos sintácticos está asociado a una PU de la CU, y cada elemento sintáctico del segundo grupo de elementos sintácticos se corresponde con un elemento sintáctico del primer grupo de elementos sintácticos. El primer grupo de elementos sintácticos y el segundo grupo de elementos sintácticos incluyen una pluralidad de elementos sintácticos. Cada elemento del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva (por ejemplo, la PU que corresponde al elemento sintáctico del primer grupo de elementos sintácticos) se infiere a partir de una PU vecina. Dicho de otro modo, cada elemento del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva está basado en un índice hacia una lista de modos más probables.

10

15

Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar configurados para generar la lista de modos más probables en la manera de ejemplo descrita anteriormente. Cada elemento sintáctico del primer grupo de elementos sintácticos puede corresponder a una de las PU de la CU. En este ejemplo, un elemento sintáctico del primer grupo de elementos sintácticos (es decir, uno de los elementos sintácticos de la pluralidad de elementos sintácticos del primer grupo) indica si el modo de intra-predicción para la PU a la que corresponde el elemento sintáctico en la CU está basado en un índice hacia la lista de modos más probables.

20

25

Si el elemento sintáctico del primer grupo de elementos sintácticos indica que el modo de intra-predicción de la PU, a la que está asociado, está basado en el índice hacia la lista de modos más probables, entonces un elemento sintáctico del segundo grupo de elementos sintácticos, que está asociado al elemento sintáctico del primer grupo de elementos sintácticos, identifica el índice hacia la lista de modos más probables. Si el elemento sintáctico del primer grupo de elementos sintácticos indica que el modo de intra-predicción de la PU, a la que está asociado, no está basado en el índice hacia la lista de modos más probables, entonces un elemento sintáctico del segundo grupo de elementos sintácticos, que está asociado al elemento sintáctico del primer grupo de elementos sintácticos, indica el modo de intra-predicción.

30

35

La FIG. 3 es un diagrama de bloques que ilustra un ejemplo del codificador de vídeo 20, que puede implementar las técnicas de codificación de datos de vídeo descritas en el presente documento. En un ejemplo, el codificador de vídeo 20 puede estar configurado para generar un primer grupo de elementos sintácticos y un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde cada uno de los elementos sintácticos del primer grupo de elementos sintácticos indica si un modo de intra-predicción de una PU respectiva en la CU está basado en un índice hacia la lista de modos más probables. El codificador de vídeo 20 puede codificar mediante codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos (por ejemplo, usando un motor CABAC convencional); después de codificar por entropía el primer grupo de elementos sintácticos, codificar por circunvalación el segundo grupo de elementos sintácticos (por ejemplo, usando un motor de codificación por circunvalación) y proporcionar los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos. El primer y el segundo grupo de elementos sintácticos incluyen una pluralidad de elementos sintácticos.

40

45

50

El codificador de vídeo 20 puede llevar a cabo la intra-codificación (es decir, la codificación de intra-predicción) y la inter-codificación (es decir, la codificación de inter-predicción) de bloques de vídeo dentro de fragmentos de vídeo. Como se muestra en la FIG. 3, el codificador de vídeo 20 recibe datos de vídeo, que pueden ser un bloque de vídeo actual de un cuadro de vídeo que va a codificarse, y proporciona un flujo de bits de vídeo codificado. Además, el codificador de vídeo 20 puede enviar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en cuadros y datos sintácticos basados en GOP, por ejemplo en una cabecera de cuadro, una cabecera de bloque, una cabecera de fragmento o una cabecera de GOP. Los datos sintácticos de GOP pueden describir un número de cuadros en el GOP respectivo, y los datos sintácticos de cuadro pueden indicar un modo de codificación / predicción usado para codificar el cuadro correspondiente.

55

60

En el ejemplo de la FIG. 3, el codificador de vídeo 20 incluye una unidad de selección de modo 40, una memoria de cuadros de referencia 64, un sumador 50, una unidad de procesamiento de transformada 52, una unidad de procesamiento de cuantificación 54 y una unidad de codificación de entropía 56. A su vez, la unidad de selección de modo 40 incluye una unidad de estimación de movimiento 42, una unidad de compensación de movimiento 44, una unidad de intra-predicción 46 y una unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además una unidad de procesamiento de cuantificación inversa 58, una unidad de procesamiento de transformada inversa 60 y un sumador 62.

65

Durante el proceso de codificación, el codificador de vídeo 20 recibe un cuadro o un fragmento de vídeo que va a codificarse. El cuadro o el fragmento puede estar dividido en múltiples bloques de vídeo. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 llevan a cabo una codificación de inter-predictiva (codificación de inter-predicción) del bloque de vídeo recibido con respecto a uno o más bloques de uno o más cuadros de referencia para proporcionar una predicción temporal. Como alternativa, la unidad de intra-predicción 46 puede llevar a cabo una codificación intra-predictiva (codificación de intra-predicción) del bloque de vídeo recibido con respecto a uno o más bloques vecinos del mismo cuadro o fragmento que el bloque que va a codificarse para proporcionar predicción espacial. El codificador de vídeo 20 puede llevar a cabo múltiples pasadas de codificación, por ejemplo para seleccionar un modo de codificación apropiado para cada bloque de datos de vídeo.

Además, la unidad de división 48 puede dividir bloques de datos de vídeo en subbloques, en función de la evaluación de esquemas de división anteriores en las pasadas de codificación anteriores. Por ejemplo, la unidad de división 48 puede dividir inicialmente un cuadro o un fragmento en varias LCU y dividir cada una de las LCU en varias sub-CU basándose en un análisis de velocidad-distorsión (por ejemplo, optimización de velocidad-distorsión). La unidad de selección de modo 40 puede producir además una estructura de datos de árbol cuaternario que indica la división de una LCU en varias sub-CU. Las CU de nodo hoja del árbol cuaternario pueden incluir una o más PU y una o más TU.

La unidad de selección de modo 40 puede seleccionar uno de los modos de codificación (intra o inter), por ejemplo en función de los resultados de error, y proporcionar el bloque intra-codificado o inter-codificado resultante al sumador 50 para generar datos de bloque residuales y al sumador 62 para reconstruir el bloque codificado para usarse como un cuadro de referencia. La unidad de selección de modo 40 proporciona además elementos sintácticos, tales como vectores de movimiento, indicadores de intra-modo, información de división y otra información sintáctica de este tipo, a la unidad de codificación de entropía 56. Por ejemplo, la unidad de selección de modo 40 puede proporcionar los elementos sintácticos **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag**, descritos anteriormente, a la unidad de codificación de entropía 56. Además, la unidad de selección de modo 40 puede configurarse para ordenar los elementos sintácticos de la siguiente manera: **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode** e **intra_chroma_pred_mode**, para que se proporcionen a la unidad de codificación de entropía 56 de tal manera que el número de ciclos requeridos para codificar por entropía los elementos sintácticos se reduzca.

Por ejemplo, el codificador de vídeo 20, a través de la unidad de codificación de entropía 56, codifica por entropía el elemento **prev_intra_luma_pred_flag** para cada PU de una CU usando codificación CABAC. En este ejemplo, el primer grupo de elementos sintácticos incluye una pluralidad de elementos **prev_intra_luma_pred_flag** (es decir, uno para cada PU de la CU). El codificador de vídeo 20, a través de la unidad de codificación de entropía 56, codifica por entropía el elemento sintáctico **mpm_idx** usando codificación por circunvalación si el elemento **prev_intra_luma_pred_flag** correspondiente indica que el modo de intra-predicción para la PU está basado en un índice hacia una lista de modos más probables, o codifica el elemento sintáctico **rem_intra_luma_pred_mode** si el elemento **prev_intra_luma_pred_flag** correspondiente indica que el modo de intra-predicción para la PU no está basado en un índice hacia una lista de modos más probables. En este ejemplo, el segundo grupo de elementos sintácticos incluye al menos un elemento de uno o más elementos sintácticos **mpm_idx** y uno o más elementos sintácticos **rem_intra_luma_pred_mode**.

Dicho de otro modo, un ejemplo de un elemento sintáctico del primer grupo de elementos sintácticos es el elemento **prev_intra_luma_pred_flag**, un ejemplo de un elemento sintáctico del segundo grupo de elementos sintácticos es el elemento sintáctico **mpm_idx**, y otro ejemplo de un elemento sintáctico del segundo grupo de elementos sintácticos es el elemento sintáctico **rem_intra_luma_pred_mode**.

De nuevo, el elemento **prev_intra_luma_pred_flag** indica si el modo de intra-predicción de las muestras de luminancia de una PU de la CU está basado en un índice hacia una lista de modos más probables. El elemento sintáctico **mpm_idx** indica el índice hacia la lista de modos más probables si el elemento **prev_intra_luma_pred_flag** indica que el modo de intra-predicción de las muestras de luminancia de la PU de la CU está basado en un índice hacia la lista de modos más probables. El elemento sintáctico **rem_intra_luma_pred_mode** indica el modo de intra-predicción de las muestras de luminancia de la PU si el elemento **prev_intra_luma_pred_flag** indica que el modo de intra-predicción de la PU de la CU no está basado en un índice hacia la lista de modos más probables. El elemento sintáctico **rem_intra_luma_pred_mode** puede indicar un modo de intra-predicción que no está identificado en la lista de modos más probables.

La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar estrechamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, llevada a cabo por la unidad de estimación de movimiento 42, es el proceso de generar

vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo de un cuadro de vídeo o una imagen actual con respecto a un bloque predictivo de un cuadro de referencia (u otra unidad codificada) con respecto al bloque actual que está codificándose en el cuadro actual (u otra unidad codificada). Un bloque predictivo es un bloque que se corresponde estrechamente con el bloque que va a codificarse, en lo que respecta a la diferencia de píxeles, que puede determinarse mediante la suma de una diferencia absoluta (SAD), una suma de diferencia de cuadrados (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones de fracción de píxel de imágenes de referencia almacenadas en la memoria de cuadros de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones de fracción de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede llevar a cabo una búsqueda de movimiento con respecto a las posiciones de píxeles completos y las posiciones de fracción de píxel y proporcionar un vector de movimiento con una precisión de fracción de píxel.

La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento inter-codificado comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse a partir 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 de cuadros de referencia 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación de entropía 56 y a la unidad de compensación de movimiento 44.

La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar extraer o generar el bloque predictivo en función del vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden combinarse de manera funcional en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 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 está codificándose, generando valores de diferencia de píxel, como se describe posteriormente. En general, la unidad de estimación de movimiento 42 lleva a cabo una estimación de movimiento con respecto a las componentes de luminancia, y la unidad de compensación de movimiento 44 usa vectores de movimiento calculados en función de las componentes de luminancia para las componentes de crominancia y las componentes de luminancia. La unidad de selección de modo 40 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por parte del descodificador de vídeo 30 a la hora de descodificar los bloques de vídeo del fragmento de vídeo.

La unidad de intra-predicción 46 puede intra-predecir un bloque actual, como alternativa a la inter-predicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de intra-predicción 46 puede determinar un modo de intra-predicción a usar para codificar un bloque actual. En algunos ejemplos, la unidad de intra-predicción 46 puede codificar un bloque actual usando varios modos de intra-predicción, por ejemplo durante diferentes pasadas de codificación, y la unidad de intra-predicción 46 (o la unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo de intra-predicción apropiado a usar de entre los modos probados, por ejemplo en función de un análisis de velocidad-distorsión. Como se ha descrito anteriormente, los posibles modos de intra-predicción pueden incluir modos de predicción plana, una predicción DC, modos de predicción angular y un modo de predicción para una componente de crominancia que infiere un modo de intra-predicción a partir de un modo de predicción de luminancia.

Además, en un ejemplo, la unidad de intra-predicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intra-predicción probados, y seleccionar el modo de intra-predicción que tiene las mejores características de velocidad-distorsión de entre los modos probados. El análisis de velocidad-distorsión determina generalmente una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se codificó para generar el bloque codificado, así como una velocidad binaria (es decir, el número de bits) usada para generar el bloque codificado. La unidad de intra-predicción 46 puede calcular relaciones a partir de las distorsiones y velocidades de los diversos bloques codificados para determinar qué modo de intra-predicción presenta el mejor valor de velocidad-distorsión para el bloque.

Tras seleccionar un modo de intra-predicción para un bloque, la unidad de intra-predicción 46 puede proporcionar información que indica el modo de intra-predicción seleccionado para el bloque, tal como elementos sintácticos, a la unidad de codificación de entropía 56. La unidad de codificación de entropía 56 puede codificar la información que indica el modo de intra-predicción seleccionado. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido, que pueden incluir una pluralidad de

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

Como se ha descrito anteriormente, los elementos sintácticos de intra-predicción **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag** pueden indicar uno o más modos de intra-predicción de luminancia para una estructura de PU y un modo de intra-predicción de crominancia para una estructura de PU. Además, como se ha descrito anteriormente con respecto a las Tablas 3 y 4, el orden de los elementos sintácticos de intra-predicción **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag** dentro de una estructura de codificación puede determinar el número total de ciclos requeridos para codificar por entropía todos los elementos sintácticos para una estructura de PU. Por tanto, además de proporcionar los elementos sintácticos de intra-predicción a la unidad de codificación de entropía 56, como se ha descrito anteriormente con respecto a las Tablas 3 y 4, la unidad de intra-predicción 46 puede configurarse para ordenar la sintaxis de intra-predicción de modo que se proporcionen a la unidad de codificación de entropía 56 de tal manera que se reduzca el número total de ciclos requeridos para codificar por entropía los elementos sintácticos. Asimismo, con esta disposición, se reduce el número total de ciclos requeridos para descodificar por entropía los elementos sintácticos en un descodificador.

En un ejemplo, la unidad de intra-predicción 46 puede configurarse para ordenar *celldas* codificadas mediante CABAC de elementos sintácticos de intra-modo de crominancia con *celldas* codificadas mediante CABAC de elementos sintácticos de intra-modo de luminancia, de modo que se codifiquen juntas. Además, la unidad de intra-predicción 46 puede ordenar *celldas* codificadas por circunvalación de elementos sintácticos de intra-modo de luminancia con *celldas* codificadas por circunvalación de elementos sintácticos de intra-modo de crominancia, de modo que puedan codificarse juntas. La Tabla 5 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_2NX2N. La Tabla 6 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_NXN. En las Tablas 5 y 6, los elementos sintácticos pueden definirse como se ha descrito anteriormente con respecto a las Tablas 3 y 4. Sin embargo, debe observarse que los elementos sintácticos de las Tablas 5 y 6 no necesitan tener necesariamente el mismo intervalo de valores como el descrito anteriormente. Por ejemplo, si el codificador de vídeo 20 incluye un número mayor o menor que 33 modos de intra-predicción, los valores de los elementos sintácticos pueden ajustarse en consecuencia. Como se ilustra en las Tablas 5 y 6, el elemento sintáctico **chroma_mode_cabac_bins** está dispuesto de manera que se codifica antes que los elementos **mpm_idx** y **rem_intra_luma_pred_mode**.

Tabla 5 Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag [x0][y0])	
mpm_idx [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

Tabla 6 Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag [x0][y0])	
mpm_idx [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [n0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

prev_intra_luma_pred_flag[x1][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x 1][y0])	
mpm_idx[x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x0][y1]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y1])	
mpm_idx[x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x1][y1]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x 1][y1])	
mpm_idx[x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)

En otro ejemplo, la unidad de intra-predicción 46 puede configurarse para ordenar *celdas* codificadas mediante CABAC de elementos sintácticos de intra-modo de luminancia de todas las PU de una CU de manera que puedan codificarse juntas. Además, la unidad de intra-predicción 46 puede ordenar *celdas* codificadas por circunvalación de elementos sintácticos de intra-modo de luminancia de modo que puedan codificarse juntas. La Tabla 7 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_2NX2N. La Tabla 8 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_NXN. En las Tablas 7 y 8, los elementos sintácticos pueden definirse como se ha descrito anteriormente con respecto a las Tablas 5 y 6.

Como se ilustra en la Tabla 8, para una estructura de PU INTRA_NXN, se codifican los elementos sintácticos **prev_intra_luma_pred_flag** para cada una de las cuatro PU, y después se codifican los respectivos elementos sintácticos **mpm_idx** y **rem_intra_luma_pred_mode** para cada una de las PU. Los cuatro elementos sintácticos **prev_intra_luma_pred_flag** pueden denominarse primer grupo de elementos sintácticos para una PU. Por tanto, la unidad de intra-predicción 46 puede configurarse para proporcionar el primer grupo de elementos sintácticos antes que la unidad de codificación de entropía 56, antes de los elementos sintácticos respectivos **mpm_idx** y **rem_intra_luma_pred_mode** asociados a cada uno de los indicadores.

Tabla 7 Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

Tabla 8 Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag[x0][y0]	(CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x1][y0]	(CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x0][y1]	(CODIFICADO MEDIANTE CABAC)

prev_intra_luma_pred_flag [x1][y1]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y0])	
mpm_idx [x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x0][y1])	
mpm_idx [x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y1])	
mpm_idx [x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

En otro ejemplo adicional, la unidad de intra-predicción 46 puede configurarse para ordenar todas las celdas codificadas mediante CABAC de elementos sintácticos de intra-modo de luminancia y crominancia de todas las PU de una unidad de comunicación, de manera que puedan codificarse juntas. Además, la unidad de intra-predicción 46 puede configurarse para ordenar todas las celdas codificadas por circunvalación de intra-modo de luminancia y crominancia de modo que puedan codificarse juntas. La Tabla 9 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_2NX2N. La Tabla 10 proporciona una estructura de codificación de ejemplo de esta disposición para una estructura de PU INTRA_NXN. En las Tablas 9 y 10, los elementos sintácticos pueden definirse como se ha descrito anteriormente con respecto a las Tablas 5 y 6.

Tabla 9: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag [x0][y0]	(CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

Tabla 10: Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag [x0][y0]	(CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag [x1][y0]	(CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag [x0][y1]	(CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag [x1][y1]	(CODIFICADO MEDIANTE CABAC)

	MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0]	(CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y0])	
mpm_idx [x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x1][y0]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x0][y1])	
mpm_idx [x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x0][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y1])	
mpm_idx [x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
Si no	
rem_intra_luma_pred_mode [x1][y1]	(CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x0][y0]	(CODIFICADO POR CIRCUNVALACIÓN)

5 Debe observarse que en las Tablas 6 a 10, la fila en la que un elemento sintáctico se presenta en una tabla puede corresponder al orden en que se codifica por entropía mediante un codificador de entropía. En algunos casos, no puede realizarse una codificación de entropía intermedia entre filas adyacentes en las Tablas 6 a 10. Los términos "grupo" y "agrupación" pueden referirse generalmente en el presente documento a situar elementos de manera muy próxima, por ejemplo en una tabla de sintaxis de una unidad de codificación de vídeo tal como una imagen, un fragmento, una CU o similar. En casos en los que se lleva a cabo una operación en elementos sintácticos incluidos en un grupo, en algunos casos los elementos pueden situarse de manera tan próxima que la operación no se lleva a cabo en elementos externos al grupo hasta que la operación se complete en todos los elementos del grupo (es decir, sin entrelazado entre los diferentes grupos). En algunos casos, las celdas de un grupo pueden presentarse de manera consecutiva y codificarse en un orden o secuencia, es decir, una detrás de otra.

10

15 Haciendo de nuevo referencia a la FIG. 3, el codificador de vídeo 20 forma un bloque de vídeo residual sustrayendo los datos de predicción de la unidad de selección de modo 40 al bloque de vídeo original que está codificándose. El sumador 50 representa el componente o los componentes que llevan a cabo esta operación de sustracción. La unidad de procesamiento de transformada 52 aplica una transformada, tal como una transformada discreta del coseno (DCT) o una transformada conceptualmente similar, al bloque residual, generando un bloque de vídeo que comprende valores de coeficientes de transformada residuales. La unidad de procesamiento de transformada 52 puede llevar a cabo otras transformadas que son conceptualmente similares a la DCT. También pueden usarse transformadas de ondícula, transformadas de número entero, transformadas de subbanda u otros tipos de transformadas. En cualquier caso, la unidad de procesamiento de transformada 52 aplica la transformada al bloque residual, generando un bloque de

20 coeficientes de transformada residuales. La transformada puede convertir la información residual desde un dominio de valor de píxel a un dominio de transformada, tal como un dominio de frecuencia. La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de procesamiento de cuantificación 54 cuantifica los coeficientes de transformada para reducir adicionalmente la velocidad binaria. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o todos los coeficientes. El grado de cuantificación puede modificarse ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de procesamiento de cuantificación 54 puede llevar a cabo una exploración de la matriz, incluyendo los coeficientes de transformada cuantificados. Como alternativa, la unidad de codificación de entropía 56 puede llevar a cabo la exploración.

25

30

35

El codificador de vídeo 20 puede configurarse para llevar a cabo una cuantificación inversa y una transformación inversa para almacenar bloques descodificados que van a usarse como referencia para predecir bloques subsiguientes, por ejemplo en el mismo cuadro o cuadros que van a predecirse de manera temporal. La unidad de procesamiento de cuantificación inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican una cuantificación inversa y una transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo para usarse posteriormente como un bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de uno de los cuadros de la memoria de cuadros de referencia 64. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de fracción de píxel para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento generado por la unidad de compensación de movimiento 44 para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria de cuadros de referencia 64.

Tras la cuantificación, la unidad de codificación de entropía 56 codifica por entropía los coeficientes de transformada cuantificados. Además, la unidad de codificación de entropía 56 codifica por entropía elementos sintácticos, tales como los elementos sintácticos predictivos descritos anteriormente. Por ejemplo, la unidad de codificación de entropía 56 puede llevar a cabo una codificación de longitud variable adaptable al contexto (CAVLC), una codificación aritmética binaria adaptable al contexto (CABAC), una codificación aritmética binaria adaptable al contexto y basada en la sintaxis (SBAC), una codificación de entropía mediante división en intervalos de probabilidad (PIPE) u otra técnica de codificación de entropía. En el caso de la codificación de entropía basada en el contexto, un contexto puede basarse en bloques vecinos. Tras la codificación de entropía realizada por la unidad de codificación de entropía 56, el flujo de bits codificado puede transmitirse a otro dispositivo (por ejemplo, el descodificador de vídeo 30) o guardarse para su posterior transmisión o recuperación.

La FIG. 4 es un diagrama de bloques que ilustra una unidad de codificación de entropía 56 de ejemplo que puede implementar las técnicas descritas en esta divulgación. En un ejemplo, la unidad de codificación de entropía 56 ilustrada en la FIG. 4 puede ser un codificador CABAC. La unidad de codificación de entropía 56 de ejemplo puede incluir una unidad de binarización 502, una unidad de codificación aritmética 510, que incluye un motor de codificación por circunvalación 504, un motor de codificación convencional 508 y una unidad de modelización de contexto 506.

La unidad de codificación de entropía 56 puede recibir uno o más elementos sintácticos, tal como los elementos sintácticos de intra-predicción **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag**, descritos anteriormente. El orden en que los elementos sintácticos son recibidos por la unidad de codificación de entropía 56 puede definirse según una estructura de codificación, tal como las estructuras de codificación de ejemplo descritas anteriormente con respecto a las Tablas 3 a 10.

La unidad de binarización 502 recibe un elemento sintáctico y genera una cadena de *celdas* (es decir, una cadena binaria). La unidad de binarización 502 puede usar, por ejemplo, una sola o una combinación de las siguientes técnicas para generar una cadena de *celdas*: codificación de longitud fija, codificación unaria, codificación unaria truncada, codificación de Rice truncada, codificación de Golomb, codificación de Golomb exponencial y codificación de Golomb-Rice. Además, en algunos casos, la unidad de binarización 502 puede recibir un elemento sintáctico, tal como una cadena binaria, y simplemente transferir los valores de *celda*. En un ejemplo, la unidad de binarización 502 recibe el elemento sintáctico **intra_chroma_pred_mode** y genera una cadena de *celdas* basándose en el valor del elemento sintáctico **chroma_pred_from_luma_enable_flag** según el ejemplo descrito anteriormente con respecto a la Tabla 2.

La unidad de codificación aritmética 510 está configurada para recibir una cadena de *celdas* desde la unidad de binarización 502 y para llevar a cabo una codificación aritmética en la cadena de *celdas*. Como se muestra en la FIG. 4, la unidad de codificación aritmética 510 puede recibir valores de *celda* desde una ruta de circunvalación o la ruta de codificación convencional. Los valores de *celda* que siguen la ruta de circunvalación pueden ser valores de *celda* codificadas por circunvalación, y los valores de *celda* que siguen la ruta de codificación convencional pueden identificarse como codificadas mediante CABAC. En consonancia con el proceso CABAC descrito anteriormente, en caso de que la unidad de codificación aritmética 510 reciba valores de *celda* desde una ruta de circunvalación, el motor de codificación por circunvalación 504 puede llevar a cabo una codificación aritmética en los valores de *celda* sin utilizar un contexto adaptativo asignado a un valor de *celda*. En un ejemplo, el motor de codificación por circunvalación 504 puede suponer las mismas probabilidades para posibles valores de una celda.

En caso de que la unidad de codificación aritmética 510 reciba valores de *celda* a través de la ruta convencional, la unidad de modelización de contexto 506 puede proporcionar una variable de contexto (por ejemplo, un estado de contexto), de manera que el motor de codificación convencional 508 puede llevar a

cabo una codificación aritmética basándose en las asignaciones de contexto proporcionadas por la unidad de modelización de contexto 506. Las asignaciones de contexto pueden definirse según una norma de codificación de vídeo, tal como HEVC. Los modelos de contexto pueden almacenarse en memoria. La unidad de modelización de contexto 506 puede incluir una serie de tablas indexadas y/o utilizar funciones de correlación para determinar un contexto y una variable de contexto para una celda particular. Tras codificar un valor de *celda*, el motor de codificación convencional 508 puede actualizar un contexto basándose en los valores de *celda* reales.

Además, una unidad de codificación de entropía 56 de ejemplo puede configurarse para codificar *celdas* en función del orden en que recibe elementos sintácticos desde la unidad de selección de modo 40. Como se ha descrito anteriormente, el orden puede definirse según una estructura de codificación, tal como las estructuras de codificación de ejemplo descritas anteriormente con respecto a las Tablas 3 a 10. El orden en que la unidad de codificación de entropía 56 recibe elementos sintácticos dentro de un conjunto de elementos sintácticos puede determinar el número total de ciclos que requiere la unidad de codificación de entropía 56 para codificar el conjunto de elementos sintácticos.

En un ejemplo, el motor de codificación convencional 508 puede necesitar más de dos ciclos para codificar una única *celda*. Además, en un ejemplo, el motor de codificación convencional 508 puede requerir $n+M$ ciclos para codificar n *celdas*, donde M es la información de control para iniciar la cadena de procesamiento. Por lo general, M es mayor que 0. Al inicio del proceso de codificación CABAC (por ejemplo, un cambio del modo de circunvalación al modo convencional), se introduce M información de control de cadena de procesamiento. En un ejemplo, el motor de codificación por circunvalación 504 puede necesitar un ciclo para codificar un elemento sintáctico de n bits, donde n es mayor que cero.

Por tanto, el número total de ciclos que la unidad de codificación aritmética 510 requiere para codificar un conjunto de *celdas* de circunvalación y de *celdas* CABAC puede basarse en el número de veces que se introduce M información de control de cadena de procesamiento. Por ejemplo, si la unidad de codificación aritmética 510 codificara los elementos sintácticos predictivos como se dispone en la Tabla 4, la información de control de cadena de procesamiento puede introducirse cinco veces y la unidad de codificación aritmética 510 puede requerir un mínimo de $5*M$ ciclos para codificar las celdas de los elementos sintácticos. Sin embargo, si la unidad de codificación aritmética 510 codificara los elementos sintácticos predictivos como se dispone en la Tabla 8, la información de control de cadena de procesamiento solo puede introducirse dos veces y la información de control de cadena de procesamiento de la unidad de codificación aritmética 510 puede reducirse a $2*M$ ciclos.

Debe observarse que un cuadro de vídeo codificado según la norma HEVC puede incluir el orden de decenas de miles de PU. Por tanto, el orden en que los elementos sintácticos de intra-predicción están dispuestos en una estructura de codificación puede tener un efecto importante en el número de ciclos que un codificador de vídeo necesita para codificar datos de vídeo. Además, debe observarse que aunque la unidad de codificación de entropía 56 de ejemplo ilustrada en la FIG. 4 se describe pasando de una operación de codificación por circunvalación a una operación de codificación convencional, en algunos casos estas operaciones pueden llevarse a cabo en paralelo. Sin embargo, en este caso, el orden en que la unidad de codificación de entropía 56 recibe elementos sintácticos codificados por circunvalación y mediante CABAC puede seguir determinando el número total de ciclos requeridos para codificar por entropía los elementos sintácticos. La información de control se genera debido al cambio de contexto. Puesto que las celdas codificadas en el modo de circunvalación pueden dar como resultado varios elementos sintácticos diferentes para los siguientes *celdas* codificadas mediante CABAC, resulta difícil extraer con anterioridad el contexto necesario (cargar la memoria intermedia de contexto) con el fin de reducir la información de control de cadena de procesamiento.

La FIG. 5 es un diagrama de flujo que ilustra un ejemplo de codificación de datos de vídeo según las técnicas de esta divulgación. Aunque el proceso de la FIG. 5 se describe a continuación llevándose a cabo generalmente por el codificador de vídeo 20, el proceso puede llevarse a cabo por cualquier combinación formada por el codificador de vídeo 20, la unidad de selección de modo 40 y/o la unidad de codificación de entropía 56.

Como se ilustra en la FIG. 5, el codificador de vídeo 20 genera un primer grupo de elementos sintácticos (602). En un ejemplo, un elemento sintáctico del primer grupo indica si un modo de predicción de una unidad de predicción (PU) respectiva está basado en un índice hacia una lista de modos más probables. En un ejemplo, el primer grupo de elementos sintácticos incluye cuatro elementos sintácticos **prev_intra_luma_pred_flag** correspondientes a las PU de una estructura de PU INTRA_NxN. El codificador de vídeo 20 genera un segundo grupo de elementos sintácticos (604). En un ejemplo, un elemento sintáctico respectivo puede ser el elemento sintáctico **mpm_idx** o el elemento sintáctico **rem_intra_luma_pred_mode**, como se ha descrito anteriormente. Los elementos sintácticos del segundo grupo corresponden a elementos sintácticos respectivos del primer grupo.

El codificador de vídeo 20 codifica mediante CABAC el primer grupo de elementos sintácticos (por ejemplo, usando un motor CABAC convencional) (606). Después de codificar mediante CABAC el primer grupo de elementos sintácticos, el codificador de vídeo 20 codifica por circunvalación el segundo grupo de elementos sintácticos (por ejemplo, usando el motor de codificación por circunvalación) (608). El codificador de vídeo 20 puede intra-codificar las PU de la CU basándose en el primer y el segundo grupo de elementos sintácticos (610). El codificador de vídeo 20 puede proporcionar los datos de vídeo, del flujo de bits, que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos (612). El codificador de vídeo 20 también proporciona la información residual para las PU intra-codificadas en el flujo de bits.

Como se ha descrito anteriormente, ejemplos del primer grupo de elementos sintácticos incluyen una pluralidad de indicadores (por ejemplo, los elementos sintácticos **prev_intra_luma_pred_flag**). Un indicador de la pluralidad de indicadores puede ser el elemento sintáctico del primer grupo de elementos sintácticos que indica si el modo de intra-predicción de muestras de luminancia de la PU respectiva está basado en un índice hacia la lista de modos más probables.

En algunos ejemplos, el segundo grupo de elementos sintácticos incluye un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el elemento sintáctico correspondiente del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables. Por ejemplo, si el elemento sintáctico **prev_intra_luma_pred_flag** de la PU respectiva indica que el modo de intra-predicción para la PU respectiva está basado en un índice hacia una lista de modos más probables, entonces el elemento sintáctico **mpm_idx** correspondiente para la PU respectiva indica el índice hacia la lista de modos más probables. En este ejemplo, el elemento sintáctico **mpm_idx** se considera el primer elemento sintáctico del segundo grupo de elementos sintácticos. En algunos ejemplos, el segundo grupo de elementos sintácticos incluye un segundo elemento sintáctico que indica el modo de intra-predicción para la PU respectiva si el elemento sintáctico correspondiente del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables. Por ejemplo, si el elemento sintáctico **prev_intra_luma_pred_flag** de la PU respectiva indica que el modo de intra-predicción para la PU respectiva no está basado en un índice hacia una lista de modos más probables, entonces el elemento sintáctico **rem_intra_luma_pred_mode** correspondiente para la PU respectiva indica el modo de intra-predicción. En este ejemplo, el elemento sintáctico **rem_intra_luma_pred_mode** se considera el segundo elemento sintáctico del segundo grupo de elementos sintácticos.

Como se ha descrito anteriormente, la lista de modos más probables generada por el codificador de vídeo 20 identifica uno o más modos de intra-predicción de una o más PU vecinas. En algunos ejemplos, el codificador de vídeo 20 puede proporcionar todos los elementos sintácticos del primer grupo de elementos sintácticos para las PU de la CU antes de proporcionar cualquier elemento sintáctico del segundo grupo de elementos sintácticos para las PU de la CU. En algunos ejemplos, el codificador de vídeo 20 puede codificar mediante CABAC todos los elementos sintácticos del primer grupo de elementos sintácticos antes de codificar por circunvalación cualquier elemento sintáctico del segundo grupo de elementos sintácticos.

Debe observarse que, en algunos ejemplos, un codificador de entropía puede ser un codificador de entropía incluido en el codificador de vídeo 20, tal como la unidad de codificación de entropía 56. En este caso, el término 'proporcionar' puede referirse a un componente del codificador de vídeo 20 que proporciona datos a otro componente del codificador de vídeo 20. Además, en otros ejemplos, un codificador de entropía puede ser externo al codificador de vídeo 20. En un ejemplo, el codificador de vídeo 20 proporciona el conjunto de indicadores y de elementos sintácticos respectivos, de manera que los grupos de elementos sintácticos se ordenan en una secuencia antes que los elementos sintácticos respectivos. En un ejemplo, el codificador de vídeo 20 puede ordenar los elementos sintácticos según la estructura de codificación ilustrada en la Tabla 8. Un descodificador de vídeo puede recibir el flujo de bits codificado por entropía y reconstruir datos de vídeo usando el flujo de bits codificado por entropía

La FIG. 6 es un diagrama de bloques que ilustra un ejemplo del descodificador de vídeo 30, que descodifica una secuencia de vídeo codificada. En un ejemplo, el descodificador de vídeo 30 puede estar configurado para recibir un flujo de bits codificado por entropía que incluye un conjunto de indicadores y un elemento sintáctico respectivo correspondiente a cada indicador, donde cada indicador indica si un modo de predicción de una unidad de predicción respectiva (PU) está basado en un índice hacia una lista de modos más probables, descodificar mediante CABAC el conjunto de indicadores, descodificar por circunvalación los elementos sintácticos respectivos y reconstruir los datos de vídeo en función de los valores de cada indicador y de los elementos sintácticos respectivos.

En el ejemplo de la FIG. 6, el descodificador de vídeo 30 incluye una unidad de descodificación de entropía 70, una unidad de compensación de movimiento 72, una unidad de intra-predicción 74, una unidad de procesamiento de cuantificación inversa 76, una unidad de procesamiento de transformada inversa 78, una

memoria de cuadros de referencia 82 y un sumador 80. En algunos ejemplos, el descodificador de vídeo 30 puede llevar a cabo una pasada de descodificación generalmente recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 (FIG. 4).

5 La unidad de descodificación de entropía 70 recibe un flujo de bits codificado por entropía y descodifica elementos sintácticos del flujo de bits según un proceso recíproco al proceso de codificación de entropía usado para codificar los elementos sintácticos. En un ejemplo, el proceso de codificación de entropía usado para codificar los elementos sintácticos puede ser cualquiera de los procesos de codificación de entropía descritos anteriormente.

10

La FIG. 7 es un diagrama de bloques que ilustra una unidad de descodificación de entropía 70 de ejemplo que puede implementar las técnicas descritas en esta divulgación. La unidad de descodificación de entropía 70 recibe un flujo de bits codificado por entropía y descodifica elementos sintácticos del flujo de bits. Los elementos sintácticos pueden incluir los elementos sintácticos de intra-predicción **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag**, descritos anteriormente. El orden en que los elementos sintácticos se descodifican por la unidad de descodificación de entropía 70 puede definirse según una estructura de codificación, tal como las estructuras de codificación de ejemplo descritas anteriormente con respecto a las Tablas 3 a 10. La unidad de descodificación de entropía 70 de ejemplo de la FIG. 7 incluye una unidad de descodificación aritmética 702, que puede incluir un motor de descodificación por circunvalación 704 y un motor de descodificación convencional 706. La unidad de descodificación de entropía 70 de ejemplo incluye además una unidad de modelización de contexto 708 y una unidad de binarización inversa 710. La unidad de descodificación de entropía 70 de ejemplo puede llevar a cabo las funciones recíprocas de la unidad de codificación de entropía 56 de ejemplo descritas con respecto a la FIG. 4. De esta manera, la unidad de descodificación de entropía 70 puede llevar a cabo la descodificación de entropía basándose en las técnicas descritas en el presente documento.

15

20

25

30

35

La unidad de descodificación aritmética 702 recibe un flujo de bits codificado. Como se muestra en la FIG. 7, la unidad de descodificación aritmética 702 puede procesar valores de *celda* codificadas según una ruta de circunvalación o la ruta de codificación convencional. Una indicación de si un valor de *celda* codificada debería procesarse según una ruta de circunvalación o una ruta convencional puede señalizarse en el flujo de bits con una sintaxis de mayor nivel. En consonancia con el proceso CABAC descrito anteriormente, en caso de que la unidad de descodificación aritmética 702 reciba valores de *celda* desde una ruta de circunvalación, el motor de descodificación por circunvalación 704 puede llevar a cabo una codificación aritmética en los valores de *celda* sin utilizar un contexto asignado a un valor de *celda*. En un ejemplo, el motor de descodificación por circunvalación 704 puede suponer las mismas probabilidades para posibles valores de una celda.

40

45

50

En caso de que la unidad de descodificación aritmética 702 reciba valores de *celda* a través de la ruta convencional, la unidad de modelización de contexto 708 puede proporcionar una variable de contexto, de manera que el motor de descodificación convencional 706 puede llevar a cabo una codificación aritmética basándose en las asignaciones de contexto proporcionadas por la unidad de modelización de contexto 708. Las asignaciones de contexto pueden definirse según una norma de codificación de vídeo, tal como la HEVC. Los modelos de contexto pueden almacenarse en memoria. La unidad de modelización de contexto 708 puede incluir una serie de tablas indexadas y/o utilizar funciones de correlación para determinar un contexto y una parte de variable de contexto de un flujo de bits codificado. Tras descodificar un valor de *celda*, el motor de codificación convencional 706 puede actualizar un contexto basándose en los valores de *celda* descodificados. Además, la unidad de binarización inversa 710 puede llevar a cabo una binarización inversa en un valor de *celda* y usar una función de correspondencia de *celdas* para determinar si un valor de *celda* es válido. La unidad de binarización inversa 710 también puede actualizar la unidad de modelización de contexto basándose en la determinación de correspondencia. Por tanto, la unidad de binarización inversa 710 proporciona elementos sintácticos según una técnica de descodificación adaptable al contexto.

55

60

65

El orden en que la unidad de descodificación de entropía 70 recibe elementos sintácticos de un conjunto de elementos sintácticos puede determinar el número total de ciclos que la unidad de descodificación de entropía 70 requiere para codificar el conjunto de elementos sintácticos. En un ejemplo, el motor de descodificación convencional 706 puede necesitar más de dos ciclos para descodificar una única *celda*. Además, en un ejemplo, el motor de descodificación CABAC convencional 706 puede requerir $n+M$ ciclos para descodificar n *celdas*, donde M es la información de control para iniciar la cadena de procesamiento. Por lo general, M es mayor que 0. Al inicio del proceso de descodificación CABAC (por ejemplo, un cambio del modo de circunvalación al modo convencional), se introduce M información de control de cadena de procesamiento. En un ejemplo, el motor de descodificación por circunvalación 704 puede necesitar un ciclo para codificar un elemento sintáctico de n bits. Por tanto, el número total de ciclos que la unidad de descodificación aritmética 510 requiere para descodificar un conjunto de *celdas* de circunvalación y de *celdas* CABAC puede basarse en el número de veces que se introduce M información de control de cadena

de procesamiento. Por tanto, el orden en que los elementos sintácticos de intra-predicción están dispuestos en una estructura de codificación puede determinar el número de ciclos que el descodificador de vídeo 30 necesita para descodificar datos de vídeo. Además, debe observarse que aunque la unidad de descodificación de entropía 70 de ejemplo ilustrada en la FIG. 7 se describe pasando de una operación de descodificación por circunvalación a una operación de descodificación convencional, en algunos casos estas operaciones pueden llevarse a cabo en paralelo. Sin embargo, en este caso, el orden en que la

Haciendo de nuevo referencia a la FIG. 6, la unidad de compensación de movimiento 72 puede generar datos de predicción basándose en vectores de movimiento recibidos desde la unidad de descodificación de entropía 70. La unidad de compensación de movimiento 72 puede usar vectores de movimiento recibidos en el flujo de bits para identificar un bloque de predicción en cuadros de referencia de la memoria de cuadros de referencia 82. La unidad de compensación de movimiento 72 genera bloques compensados por movimiento, posiblemente llevando a cabo una interpolación basada en filtros de interpolación. Los identificadores de los filtros de interpolación que van a usarse para la estimación de movimiento con una precisión de fracción de píxel pueden incluirse en los elementos sintácticos. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación del bloque de vídeo para calcular valores interpolados para fracciones de píxeles de un bloque de referencia. La unidad de compensación de movimiento 72 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 según la información sintáctica recibida y usar los filtros de interpolación para generar bloques predictivos.

La unidad de compensación de movimiento 72 usa parte de la información sintáctica para determinar tamaños de macrobloques usados para codificar cuadros de la secuencia de vídeo codificada, información de división que describe cómo cada macrobloque de un cuadro de la secuencia de vídeo codificada está dividido, modos que indican cómo cada división está codificada, uno o más cuadros de referencia (y listas de cuadros de referencia) para cada macrobloque o división inter-codificado/a y otra información para descodificar la secuencia de vídeo codificada.

La unidad de intra-predicción 74 puede usar modos de intra-predicción recibidos en el flujo de bits para formar un bloque de predicción a partir de bloques espacialmente adyacentes. Los modos de intra-predicción pueden incluir los modos de intra-predicción descritos anteriormente. La unidad de intra-predicción 74 puede determinar un modo de intra-predicción a usar según los elementos sintácticos **prev_intra_luma_pred_flag**, **mpm_idx**, **rem_intra_luma_pred_mode**, **intra_chroma_pred_mode** y **chroma_pred_from_luma_enable_flag**, como se ha descrito anteriormente.

La unidad de cuantificación inversa 76 cuantifica de manera inversa, es decir, descuantifica, los coeficientes de bloque cuantificados proporcionados en el flujo de bits y descodificados por la unidad de descodificación de entropía 70. El proceso de cuantificación inversa puede incluir un proceso convencional, por ejemplo, como el definido por la norma de descodificación H.264. El proceso de cuantificación inversa también puede incluir el uso de un parámetro de cuantificación QP_{γ} calculado por un codificador de vídeo para cada macrobloque con el fin de determinar el grado de cuantificación y, asimismo, el grado de cuantificación inversa que debería aplicarse.

La unidad de transformada inversa 78 aplica una transformada inversa, por ejemplo una DCT inversa, una transformada inversa de número entero o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada con el fin de generar bloques residuales en el dominio de píxeles. El sumador 80 suma los bloques residuales a los bloques de predicción correspondientes generados por la unidad de compensación de movimiento 72 o por la unidad de intra-predicción 74 para generar bloques descodificados. De esta manera, el descodificador de vídeo 30 reconstruye bloques de vídeo a partir de un flujo de bits codificado.

La FIG. 8 es un diagrama de flujo que ilustra un ejemplo de descodificación de datos de vídeo según las técnicas de esta divulgación. Aunque el proceso de la FIG. 8 se describe a continuación llevándose a cabo generalmente por el descodificador de vídeo 30, el proceso puede llevarse a cabo por cualquier combinación formada por el descodificador de vídeo 30, la unidad de intra-predicción 74 y/o la unidad de descodificación de entropía 70.

Como se ilustra en la FIG. 8, el descodificador de vídeo 30 recibe un primer grupo de elementos sintácticos (802). El descodificador de vídeo 30 recibe un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos (804). Dicho de otro modo, el descodificador de vídeo 30 recibe el flujo de bits codificado por entropía que incluye el primer grupo de elementos sintácticos y el segundo grupo de elementos sintácticos para las unidades de predicción (PU) de una unidad de

codificación (CU).

En un ejemplo, un elemento sintáctico del primer grupo indica si un modo de predicción de una unidad de predicción (PU) respectiva está basado en un índice hacia una lista de modos más probables. En un ejemplo, el primer grupo incluye cuatro elementos sintácticos **prev_intra_luma_pred_flag** correspondientes a cuatro PU respectivas de una estructura de PU INTRA_NxN. En un ejemplo, un elemento sintáctico del segundo grupo de elementos sintácticos puede ser el elemento sintáctico **mpm_idx** o el elemento sintáctico **rem_intra_luma_pred_mode**, como se ha descrito anteriormente. En un ejemplo, el primer grupo de elementos sintácticos y el segundo grupo de elementos sintácticos pueden estar dispuestos según la estructura de codificación descrita con respecto a la Tabla 8. De esta manera, el orden en que el descodificador de vídeo 30 recibe y lleva a cabo operaciones en el conjunto de indicadores y de elementos sintácticos respectivos puede definirse según la estructura de codificación.

El descodificador de vídeo 30 descodifica mediante CABAC el primer grupo de elementos sintácticos (806). En un ejemplo, el descodificador de vídeo 30 descodifica mediante CABAC el primer grupo de elementos sintácticos usando un motor de codificación aritmética binaria adaptable al contexto (CABAC) convencional, tal como el motor de descodificación convencional 706 descrito con respecto a la FIG. 7. El descodificador de vídeo 30 descodifica por circunvalación el segundo grupo de elementos sintácticos (808). En un ejemplo, el descodificador de vídeo 30 puede descodificar por entropía el segundo grupo de elementos sintácticos después de descodificar por entropía el primer grupo de elementos sintácticos. Además, en un ejemplo, el descodificador de vídeo 30 puede descodificar por entropía el segundo grupo de elementos sintácticos usando un motor de descodificación por circunvalación, tal como el motor de descodificación por circunvalación 704 descrito con respecto a la FIG. 7. El descodificador de vídeo 30 descodifica y reconstruye datos de vídeo basándose en el primer grupo descodificado de elementos sintácticos y en el segundo grupo descodificado de elementos sintácticos (810). En un ejemplo, el descodificador de vídeo 30 reconstruye datos de vídeo sumando bloques residuales de datos de vídeo y bloques de predicción correspondientes de datos de vídeo asociados a los elementos sintácticos respectivos. Por ejemplo, el descodificador de vídeo 30 descodifica mediante intra-predicción las PU de la CU basándose en el primer grupo de elementos sintácticos y en el segundo grupo descodificado de elementos sintácticos.

Cabe señalar que, de manera similar a lo descrito anteriormente con respecto al codificador de vídeo 20 y la FIG. 5, ejemplos del primer grupo de elementos sintácticos incluyen una pluralidad de indicadores (por ejemplo, los elementos sintácticos **prev_intra_luma_pred_flag**). Un indicador de la pluralidad de indicadores puede ser el elemento sintáctico del primer grupo de elementos sintácticos que indica si el modo de intra-predicción de muestras de luminancia de la PU respectiva está basado en un índice hacia la lista de modos más probables.

En algunos ejemplos, el segundo grupo de elementos sintácticos incluye un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el elemento sintáctico correspondiente del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables. Por ejemplo, si el elemento sintáctico **prev_intra_luma_pred_flag** de la PU respectiva indica que el modo de intra-predicción para la PU respectiva está basado en un índice hacia una lista de modos más probables, entonces el elemento sintáctico **mpm_idx** correspondiente para la PU respectiva indica el índice hacia la lista de modos más probables. En este ejemplo, el elemento sintáctico **mpm_idx** se considera el primer elemento sintáctico del segundo grupo de elementos sintácticos.

En algunos ejemplos, el segundo grupo de elementos sintácticos incluye un segundo elemento sintáctico que indica el modo de intra-predicción para la PU respectiva si el elemento sintáctico correspondiente del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables. Por ejemplo, si el elemento sintáctico **prev_intra_luma_pred_flag** de la PU respectiva indica que el modo de intra-predicción para la PU respectiva no está basado en un índice hacia una lista de modos más probables, entonces el elemento sintáctico **rem_intra_luma_pred_mode** correspondiente para la PU respectiva indica el modo de intra-predicción. En este ejemplo, el elemento sintáctico **rem_intra_luma_pred_mode** se considera el segundo elemento sintáctico del segundo grupo de elementos sintácticos.

Como se ha descrito anteriormente, la lista de modos más probables generada por el descodificador de vídeo 30 identifica uno o más modos de intra-predicción de una o más PU vecinas. En algunos ejemplos, el descodificador de vídeo 30 puede recibir todos los elementos sintácticos del primer grupo de elementos sintácticos para las PU de la CU antes de recibir cualquier elemento sintáctico del segundo grupo de elementos sintácticos para las PU de la CU. En algunos ejemplos, el descodificador de vídeo 30 puede descodificar mediante CABAC todos los elementos sintácticos del primer grupo de elementos sintácticos antes de descodificar por circunvalación cualquier elemento sintáctico del segundo grupo de elementos sintácticos.

En los ejemplos anteriores, cuando el modo de predicción para una CU es INTRA_NxN, la CU incluye cuatro PU, y cada PU presenta un modo de intra-predicción de luminancia y solamente un modo de intra-predicción de crominancia para toda la CU. Sin embargo, puede haber algunos inconvenientes con un diseño de este tipo. Por ejemplo, el rendimiento de predicción puede ser ineficiente ya que cuatro PU con diferentes patrones de textura se ven obligadas a usar un modo de predicción de crominancia idéntico. Como otro ejemplo, puede haber una señalización de modo inconsistente entre las componentes de luminancia y de crominancia.

A continuación se describen algunos ejemplos para abordar los inconvenientes asociados al requisito de que haya solamente un modo de intra-predicción de crominancia para toda la CU. Las siguientes técnicas pueden aplicarse junto con las técnicas descritas anteriormente para cuando haya solamente un modo de intra-predicción de crominancia para toda la CU, o de manera independiente a las técnicas descritas anteriormente para cuando haya solamente un modo de intra-predicción de crominancia de toda la CU.

Para abordar el posible problema relacionado con un rendimiento de predicción ineficiente, mencionado anteriormente, las técnicas de esta divulgación incluyen señalar un modo de intra-predicción de crominancia para cada PU como se hace en el modo de intra-predicción de luminancia, es decir, se señalizan cuatro modos de intra-predicción de crominancia para una CU con el modo de predicción INTRA_NxN en el contexto de la norma HEVC actual. Con este diseño, un elemento sintáctico "luma_intra_prediction_mode" y un elemento sintáctico "chroma_intra_prediction_mode" se señalizan consecuentemente para representar el modo de intra-predicción para una PU.

En algunos ejemplos, este esquema de señalización intra-modo puede aplicarse a menos que el tamaño de CU actual sea igual a 8x8 y el tamaño de unidad de transformada más pequeño sea de 4x4. De manera más genérica, esta excepción existe cuando el tamaño de CU actual es dos veces el tamaño de unidad de transformada más pequeño en cada dimensión, lo que significa que una componente de crominancia de la CU actual es igual al tamaño de unidad de transformada más pequeño y no puede dividirse adicionalmente en cuatro PU (suponiendo que el formato origen sea YUV420). En tal caso, solo se señala un modo de intra-predicción de crominancia para la CU actual.

Según estos ejemplos, la estructura de codificación para los intra-modos puede implementarse como se muestra en los ejemplos de la Tabla 11 y la Tabla 12

Tabla 11: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0]	CABAC
si(prev_intra_luma_pred_flag[x0][y0])	
mpm_idx[x0][y0]	Circunvalación
Si no	n
rem_intra_luma_pred_mode[x0][y0]	Circunvalación
chroma_mode_cabac_bins [x0][y0]	CABAC
chroma_mode_bypass_bins[x0][y0]	Circunvalación
	n

Tabla 12: Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x1][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y0])
mpm_idx[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no

rem_intra_luma_pred_mode[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_cabac_bins [x1][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
}
prev_intra_luma_pred_flag[x0][y1] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y1])
mpm_idx[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_cabac_bins [x0][y1] (CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
}
prev_intra_luma_pred_flag[x1][y1] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y1])
mpm_idx[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_cabac_bins [x1][y1] (CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
}

5 A continuación se describen ejemplos de dónde está definido el modo de intra-predicción para las muestras de luminancia de cada PU y para las muestras de crominancia de cada PU de una CU, en lugar de un solo modo de intra-predicción para las muestras de crominancia de toda la CU. Las Tablas 13 y 14 definen una manera de ordenar la sintaxis de intra-predicción para la estructura de codificación intra-modo para intra_2Nx2N e intra_NxN, respectivamente, donde todas las celdas codificadas mediante CABAC del modo de intra-predicción de luminancia y del modo de intra-predicción de crominancia se codifican juntas y después todas las celdas codificadas por circunvalación del modo de intra-predicción de luminancia y del modo de intra-predicción de crominancia se codifican juntas. Las Tablas 15 y 16 definen una manera de ordenar la sintaxis de intra-predicción para la estructura de codificación intra-modo para intra_2Nx2N e intra_NxN, respectivamente, donde todas las celdas codificadas mediante CABAC del modo de intra-predicción para todas las PU de la CU se codifican juntas y después todas las celdas codificadas por circunvalación del modo de intra-predicción se codifican juntas. Las Tablas 17 y 18 definen una manera de ordenar la sintaxis de intra-predicción para la estructura de codificación intra-modo para intra_2Nx2N e intra_NxN, respectivamente, donde se codifican todas las celdas codificadas mediante CABAC del modo de intra-predicción de luminancia y crominancia para todas las PU de una CU y después todas las celdas codificadas por circunvalación del modo de intra-predicción de luminancia y crominancia se codifican juntas.

Tabla 13: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)

chroma_mode_bypass_bins[x0][y0] (CODIFICADOR POR CIRCUNVALACIÓN)

Tabla 14: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x1][y0] (CODIFICADO MEDIANTE CABAC)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_cabac_bins [x1][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y0])
mpm_idx[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_bypass_bins[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x0][y1] (CODIFICADO MEDIANTE CABAC)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_cabac_bins [x0][y1] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y1])
mpm_idx[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_bypass_bins[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
prev_intra_luma_pred_flag[x1][y1] (CODIFICADO MEDIANTE CABAC)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_cabac_bins [x1][y1] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x1][y1])
mpm_idx[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño)
chroma_mode_bypass_bins[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)

Tabla 15: Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)

Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)

Tabla 16: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x1][y0] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x0][y1] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x1][y1] (CODIFICADO MEDIANTE CABAC)
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y0])
mpm_idx[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x0][y1])
mpm_idx[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y1])
mpm_idx[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_cabac_bins [x1][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y1] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x1][y1] (CODIFICADO MEDIANTE CABAC)
}
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_bypass_bins [x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
}

Tabla 17: Estructura de codificación intra-modo para intra_NxN

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)

si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)

Tabla 18: Estructura de codificación intra-modo para intra_2Nx2N

prev_intra_luma_pred_flag[x0][y0] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x1][y0] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x0][y1] (CODIFICADO MEDIANTE CABAC)
prev_intra_luma_pred_flag[x1][y1] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y0] (CODIFICADO MEDIANTE CABAC)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_cabac_bins [x1][y0] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x0][y1] (CODIFICADO MEDIANTE CABAC)
chroma_mode_cabac_bins [x1][y1] (CODIFICADO MEDIANTE CABAC)
}
si(prev_intra_luma_pred_flag[x0][y0])
mpm_idx[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y0])
mpm_idx[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x0][y1])
mpm_idx[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
si(prev_intra_luma_pred_flag[x1][y1])
mpm_idx[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
Si no
rem_intra_luma_pred_mode[x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins[x0][y0] (CODIFICADO POR CIRCUNVALACIÓN)
si(tamaño_CU != 2*tamaño_TU_másPequeño) {
chroma_mode_bypass_bins [x1][y0] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x0][y1] (CODIFICADO POR CIRCUNVALACIÓN)
chroma_mode_bypass_bins [x1][y1] (CODIFICADO POR CIRCUNVALACIÓN)
}

5 En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse en o transmitirse como una o más instrucciones o código en un medio legible por ordenador y ejecutarse

mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como medios de almacenamiento de datos o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo según un protocolo de comunicaciones. De esta manera, los medios legibles por ordenador pueden corresponder, generalmente, a (1) medios de almacenamiento tangibles y legibles por ordenador 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 que pueda ser accedido por uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

A modo de ejemplo, y no de manera limitativa, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión puede denominarse de manera apropiada medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debe entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, se refieren a medios de almacenamiento tangibles no transitorios. Los discos, tal y como se usan en el presente documento, incluyen discos compactos (CD), discos de láser, discos ópticos, discos versátiles digitales (DVD), discos flexibles y discos blu-ray, donde los discos normalmente reproducen datos de manera magnética así como de manera óptica con láser. Las combinaciones de lo anterior también deben incluirse dentro del alcance de los medios legibles por ordenador.

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 de aplicación específica (ASIC), matrices lógicas de campo programable (FPGA), u otro circuito lógico integrado o discreto equivalente. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquier estructura anterior 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 en hardware dedicado y/o módulos de software configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Además, las técnicas pueden implementarse completamente en uno o más circuitos o elementos lógicos.

Las técnicas de esta divulgación se pueden implementar en una gran variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Varios componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas descritas, pero no requieren necesariamente la realización mediante diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por una colección de unidades de hardware interoperativas, incluyendo uno o más procesadores como se ha descrito anteriormente, junto con el software y / o firmware adecuado. Se han descrito varios ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

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

5 recibir (802, 804) un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos que comprende una pluralidad de indicadores y un segundo grupo de elementos sintácticos para unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un
10 indicador de la pluralidad de indicadores del primer grupo de elementos sintácticos indica si un modo de intra-predicción de muestras de luminancia de una PU respectiva de la CU está basado en un índice hacia una lista de modos más probables;

15 descodificar (806) según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de descodificar mediante CABAC el primer grupo de elementos sintácticos, descodificar por circunvalación (808) el segundo grupo de elementos sintácticos, donde el segundo grupo de elementos sintácticos comprende uno de:

20 un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables; y

25 un segundo elemento sintáctico que indica el modo de intra-predicción de las muestras de luminancia de la PU respectiva si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables; y

30 reconstruir (810) los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

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

35 descodificar mediante CABAC un primer conjunto de *celdas* usadas para indicar un modo de intra-predicción de crominancia separadamente de la descodificación CABAC (806) del primer grupo de elementos sintácticos; y

40 descodificar por circunvalación un segundo conjunto de *celdas* usadas para indicar el modo de intra-predicción de crominancia separadamente de la descodificación por circunvalación (808) del segundo grupo de elementos sintácticos.

45 3. El procedimiento según la reivindicación 1, en el que la recepción comprende recibir todos los elementos sintácticos del primer grupo de elementos sintácticos para las PU de la CU antes de recibir cualquier elemento sintáctico del segundo grupo de elementos sintácticos para las PU de la CU.

50 4. El procedimiento según la reivindicación 1, en el que la descodificación CABAC (806) comprende descodificar mediante CABAC todos los elementos sintácticos del primer grupo de elementos sintácticos antes de descodificar por circunvalación cualquier elemento sintáctico del segundo grupo de elementos sintácticos.

5. El procedimiento según la reivindicación 1, en el que la reconstrucción comprende:

55 descodificar mediante intra-predicción las PU de la CU en función del primer grupo de elementos sintácticos y del segundo grupo de elementos sintácticos.

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

60 generar (602) un primer grupo de elementos sintácticos que comprende una pluralidad de indicadores, donde un indicador de la pluralidad de indicadores del primer grupo de elementos sintácticos indica si un modo de intra-predicción de muestras de luminancia de una unidad de predicción (PU) respectiva de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; generar (604) un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde el segundo
65 grupo de elementos sintácticos comprende uno de:

- 5 un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables; y
- 10 un segundo elemento sintáctico que indica el modo de intra-predicción de las muestras de luminancia de la PU respectiva si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables;
- 15 codificar (606) según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; después de codificar mediante CABAC el primer grupo de elementos sintácticos, codificar por circunvalación (608) el segundo grupo de elementos sintácticos; y
- 20 proporcionar (612) los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.
- 20 7. El procedimiento según la reivindicación 6, que comprende además:
- 25 codificar mediante CABAC un primer conjunto de *celdas* usadas para indicar un modo de intra-predicción de crominancia separadamente de la codificación CABAC del primer grupo de elementos sintácticos; y
- 25 codificar por circunvalación un segundo conjunto de *celdas* usadas para indicar el modo de intra-predicción de crominancia separadamente de la codificación por circunvalación del segundo grupo de elementos sintácticos.
- 30 8. El procedimiento según la reivindicación 6, en el que la provisión comprende proporcionar los datos de vídeo que incluyen todos los elementos sintácticos del primer grupo codificado de elementos sintácticos antes de proporcionar cualquier elemento sintáctico del segundo grupo de elementos sintácticos.
- 35 9. El procedimiento según la reivindicación 6, en el que la codificación CABAC comprende codificar mediante CABAC todos los elementos sintácticos del primer grupo de elementos sintácticos antes de codificar por circunvalación (608) cualquier elemento sintáctico del segundo grupo de elementos sintácticos.
- 40 10. El procedimiento según la reivindicación 6, que comprende además:
- 40 codificar mediante intra-predicción (610) las PU de la CU en función del primer grupo de elementos sintácticos y del segundo grupo de elementos sintácticos.
- 45 11. El procedimiento según la reivindicación 1 o la reivindicación 6, en el que la lista de modos más probables identifica uno o más modos de intra-predicción de una o más PU vecinas.
- 50 12. El procedimiento según la reivindicación 1 o la reivindicación 6, en el que el primer grupo de elementos sintácticos incluye cuatro indicadores, y donde cada uno de los cuatro indicadores está asociado a una PU respectiva de una estructura de PU INTRA NXN de la CU.
- 55 13. Un medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo lleven a cabo el procedimiento según una cualquiera de las reivindicaciones anteriores.
- 60 14. Un aparato (30) para descodificar datos de vídeo, comprendiendo el aparato:
- 65 medios para recibir un flujo de bits codificado por entropía que incluye un primer grupo de elementos sintácticos que comprende una pluralidad de indicadores y un segundo grupo de elementos sintácticos para unidades de predicción (PU) de una unidad de codificación (CU), donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, y donde un indicador de la pluralidad de indicadores del primer grupo de elementos sintácticos indica si un modo de intra-predicción de muestras de luminancia de una PU respectiva de la CU está basado en un índice hacia una lista de modos más probables;

medios para descodificar según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; medios para descodificar por circunvalación el segundo grupo de elementos sintácticos, después de descodificar mediante CABAC el primer grupo de elementos sintácticos, donde el segundo grupo de elementos sintácticos comprende uno de:

5

un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables; y

10

un segundo elemento sintáctico que indica el modo de intra-predicción de las muestras de luminancia de la PU respectiva si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables; y

15

medios para reconstruir los datos de vídeo en función del primer grupo descodificado de elementos sintácticos y del segundo grupo descodificado de elementos sintácticos.

15. Un aparato (20) para codificar datos de vídeo, comprendiendo el aparato:

20

medios para generar un primer grupo de elementos sintácticos que comprende una pluralidad de indicadores, donde un indicador de la pluralidad de indicadores del primer grupo de elementos sintácticos indica si un modo de intra-predicción de muestras de luminancia de una unidad de predicción (PU) respectiva de una unidad de codificación (CU) está basado en un índice hacia una lista de modos más probables; medios para generar un segundo grupo de elementos sintácticos, donde los elementos sintácticos del segundo grupo de elementos sintácticos corresponden a elementos sintácticos respectivos del primer grupo de elementos sintácticos, donde el segundo grupo de elementos sintácticos comprende uno de:

25

un primer elemento sintáctico que indica el índice hacia la lista de modos más probables si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva está basado en el índice hacia la lista de modos más probables; y

30

un segundo elemento sintáctico que indica el modo de intra-predicción de las muestras de luminancia de la PU respectiva si el indicador del primer grupo de elementos sintácticos indica que el modo de intra-predicción de las muestras de luminancia de la PU respectiva no está basado en el índice hacia la lista de modos más probables;

35

medios para codificar según la codificación aritmética binaria adaptable al contexto (CABAC) el primer grupo de elementos sintácticos; medios (504) para codificar por circunvalación el segundo grupo de elementos sintácticos, después de codificar mediante CABAC el primer grupo de elementos sintácticos; y

40

45

medios para proporcionar los datos de vídeo que incluyen el primer grupo codificado de elementos sintácticos y el segundo grupo codificado de elementos sintácticos.

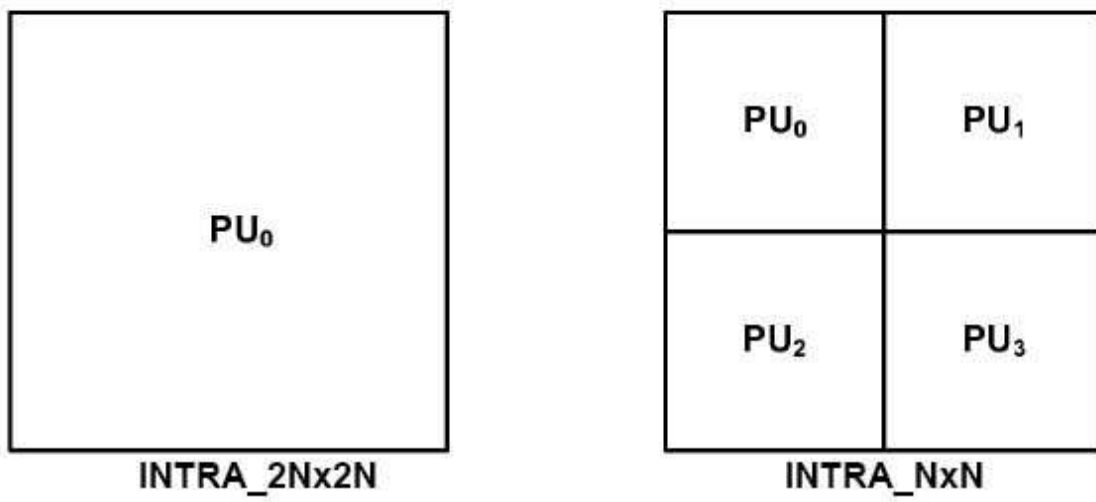


FIG. 1

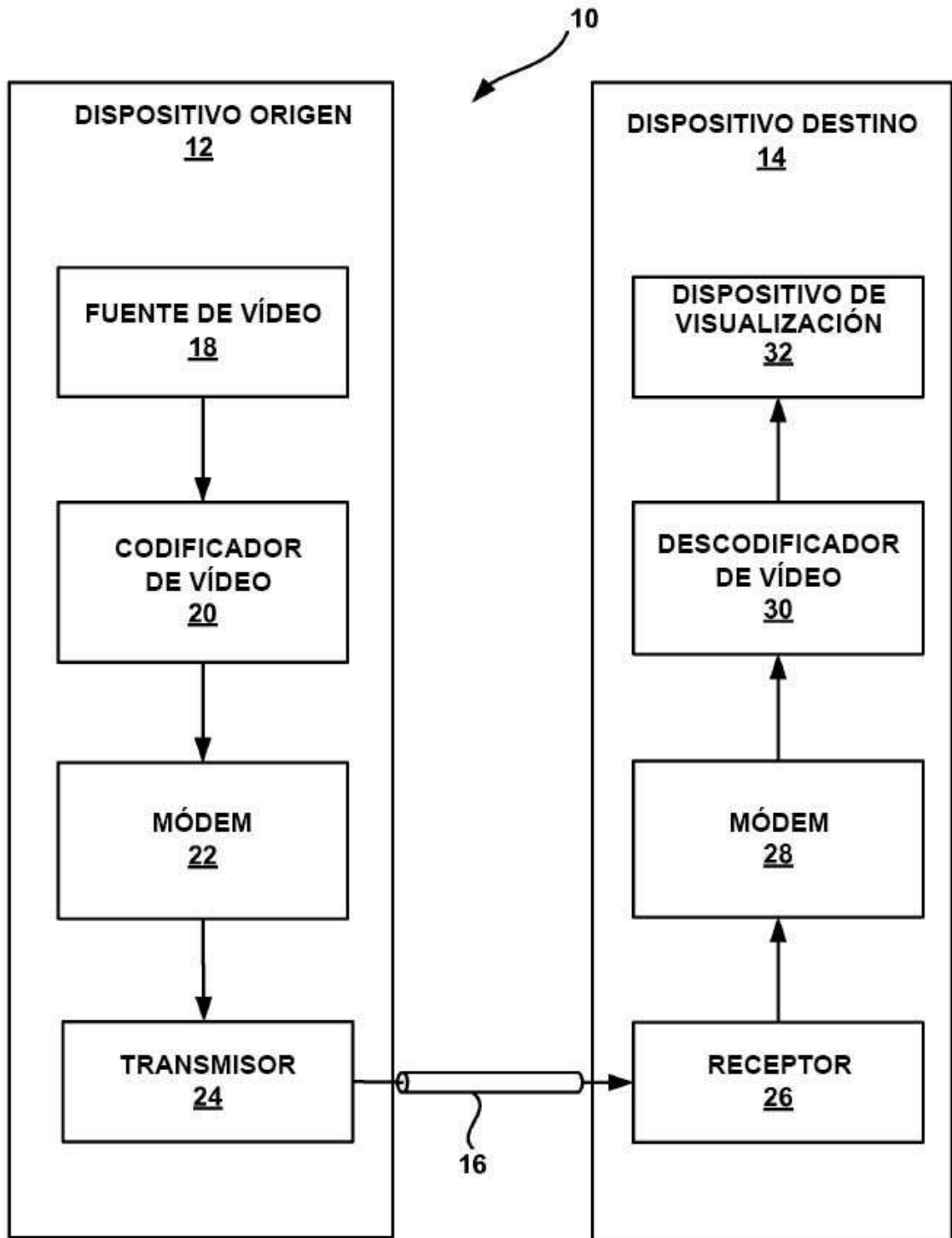


FIG. 2

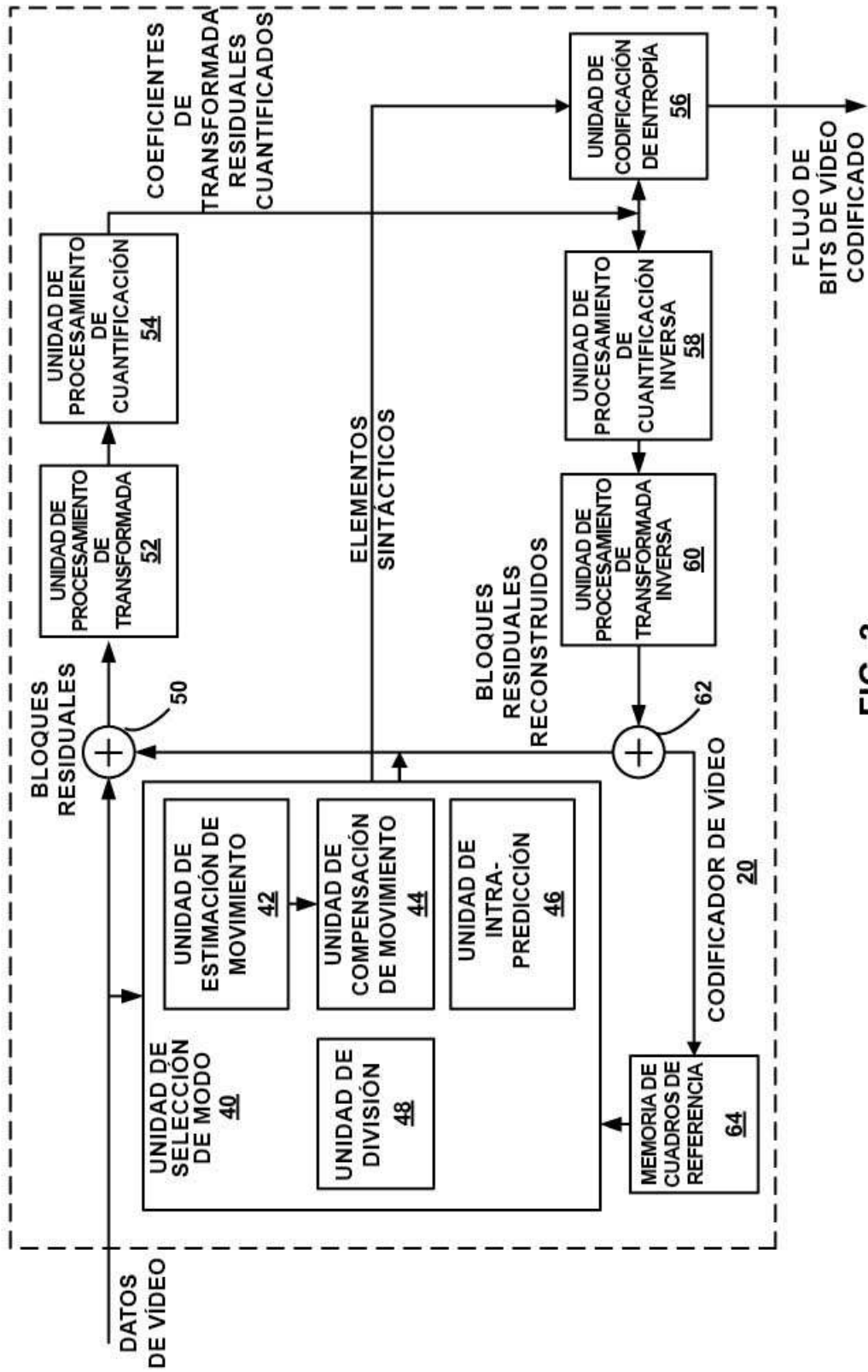


FIG. 3

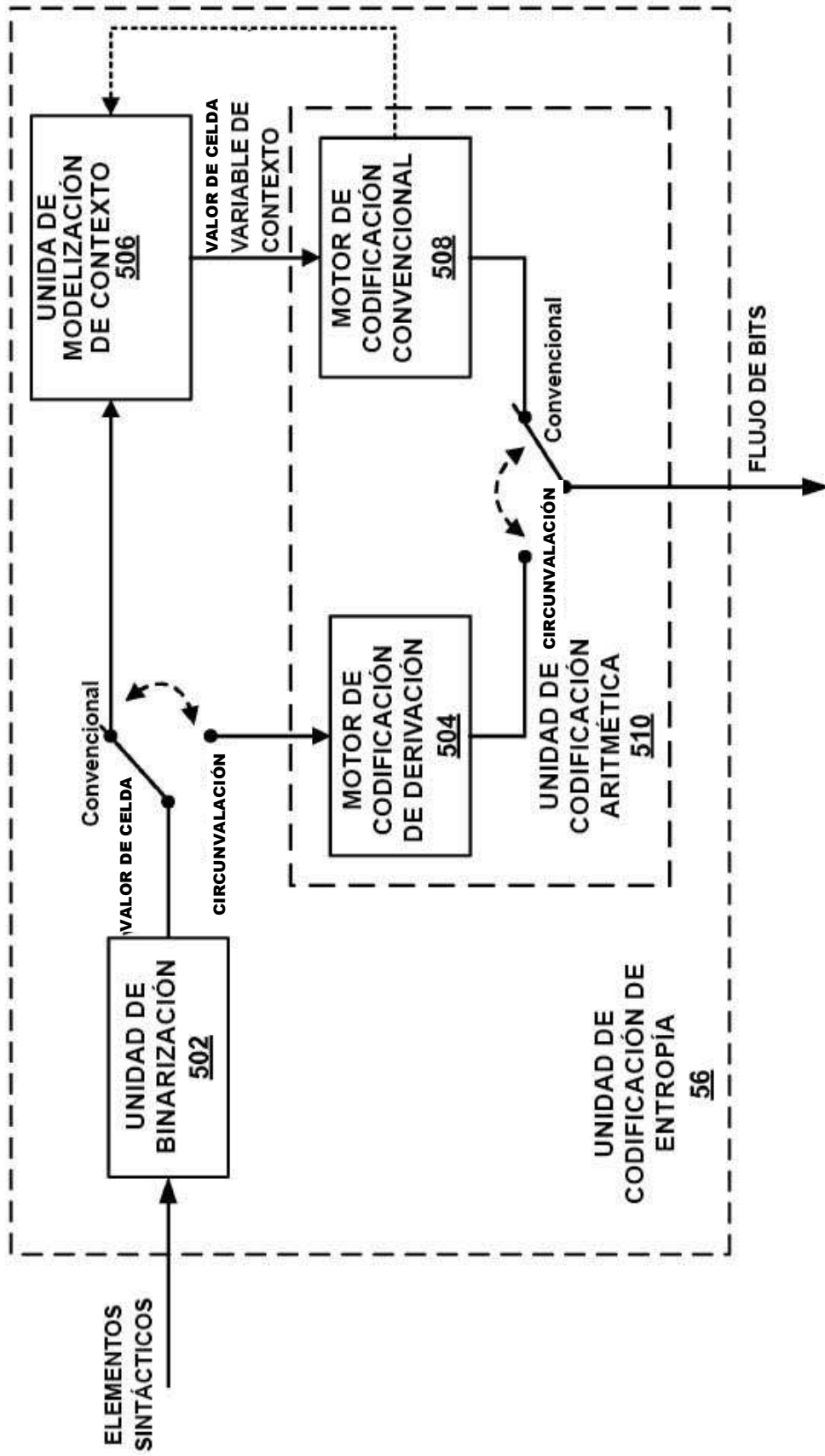


FIG. 4

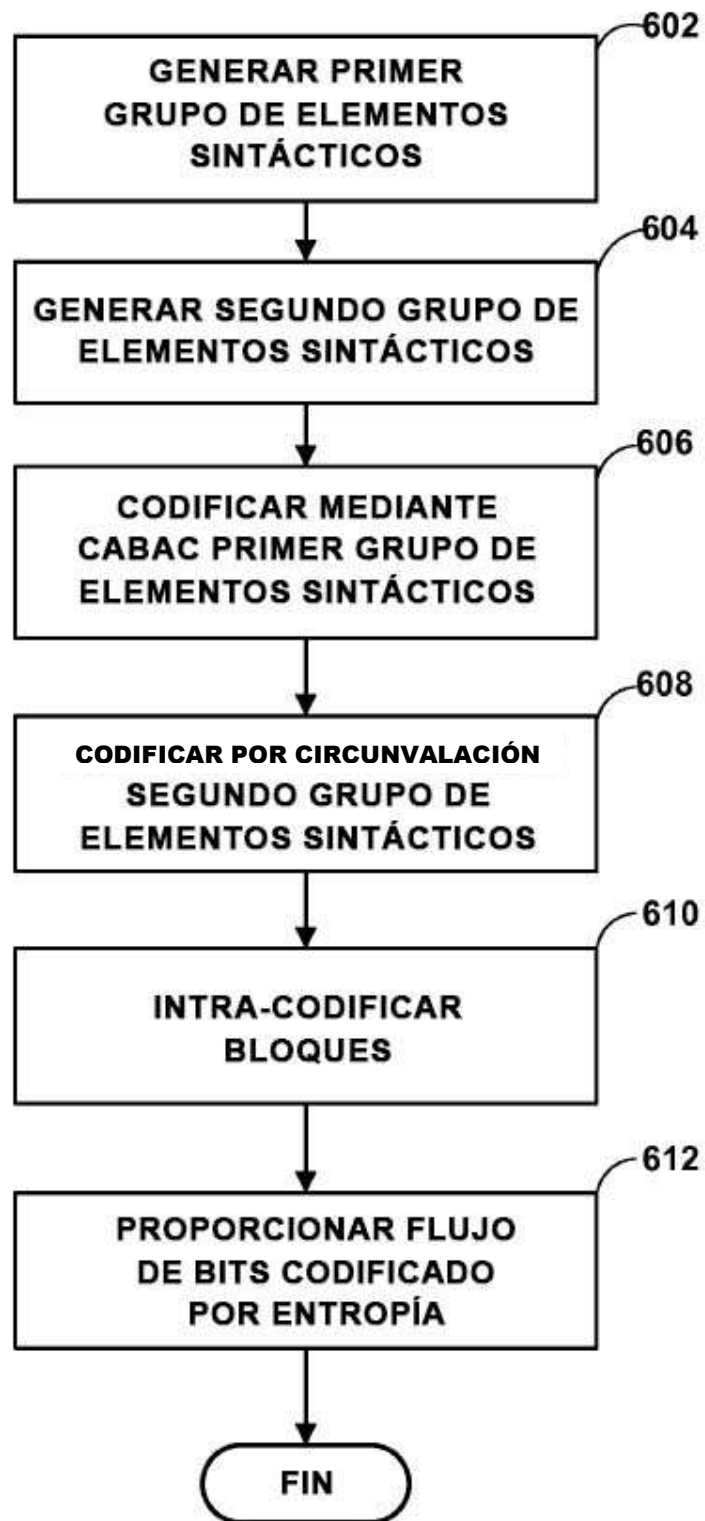
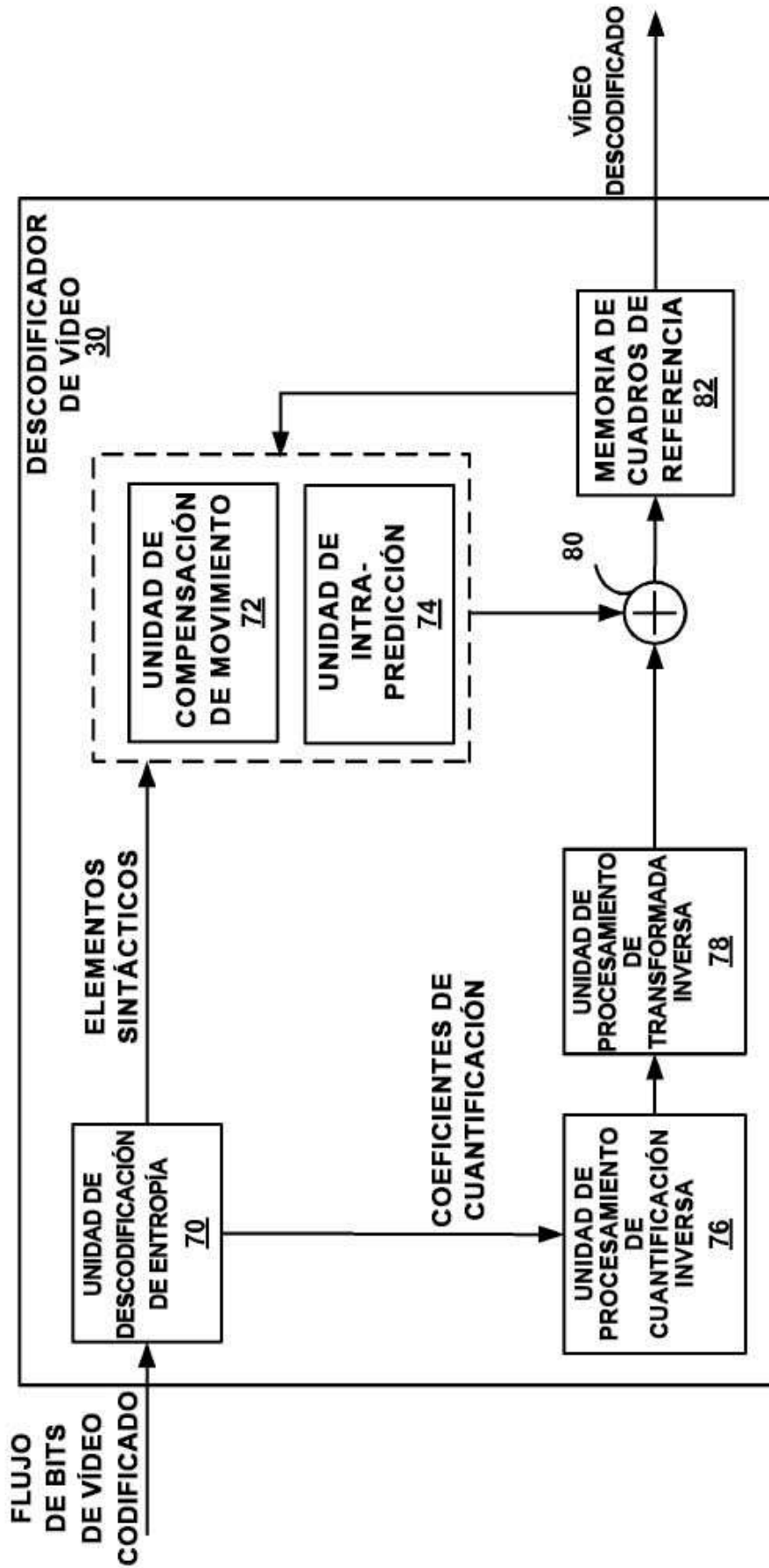


FIG. 5



BLOQUES RESIDUALES

FIG. 6

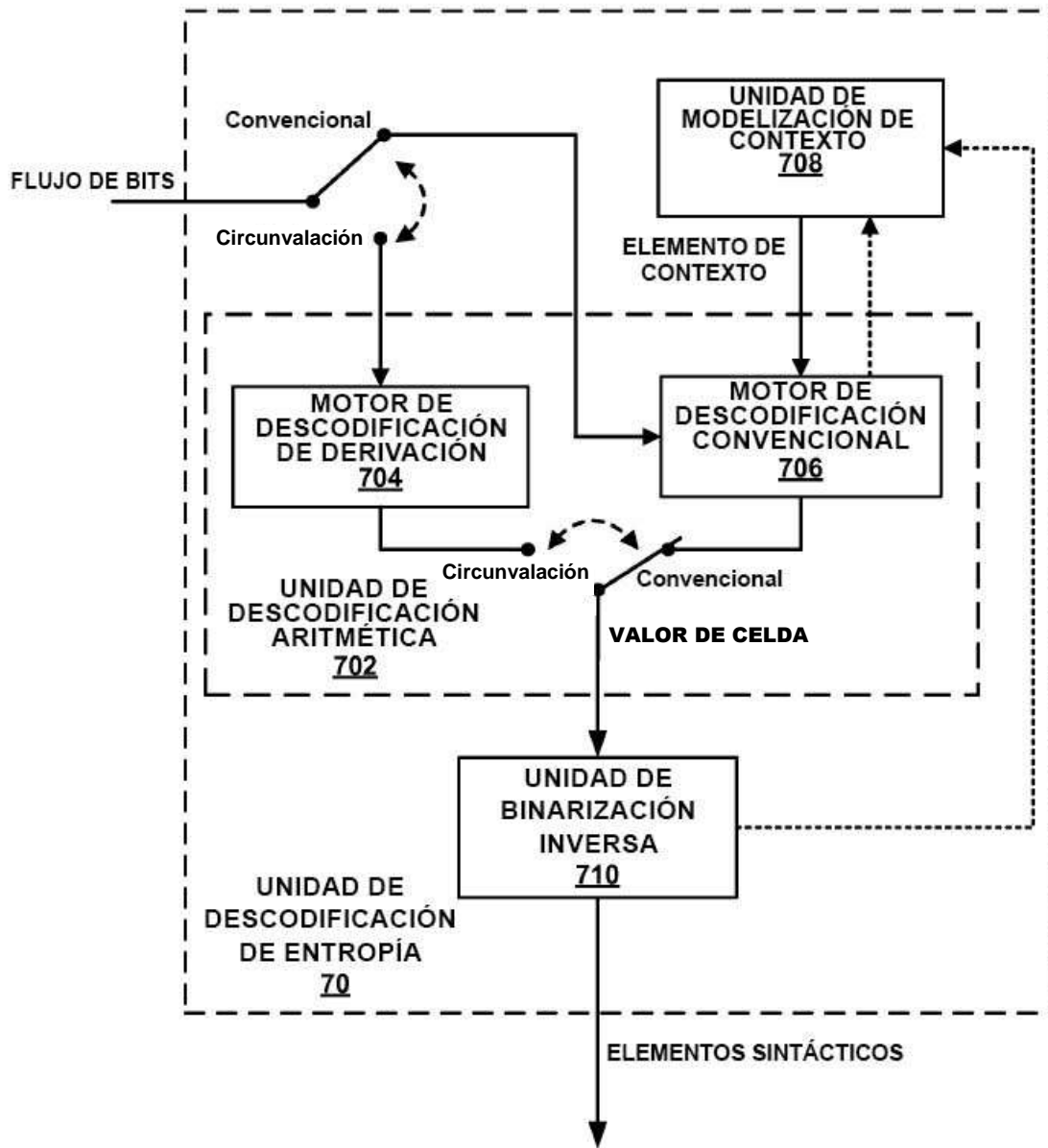


FIG. 7

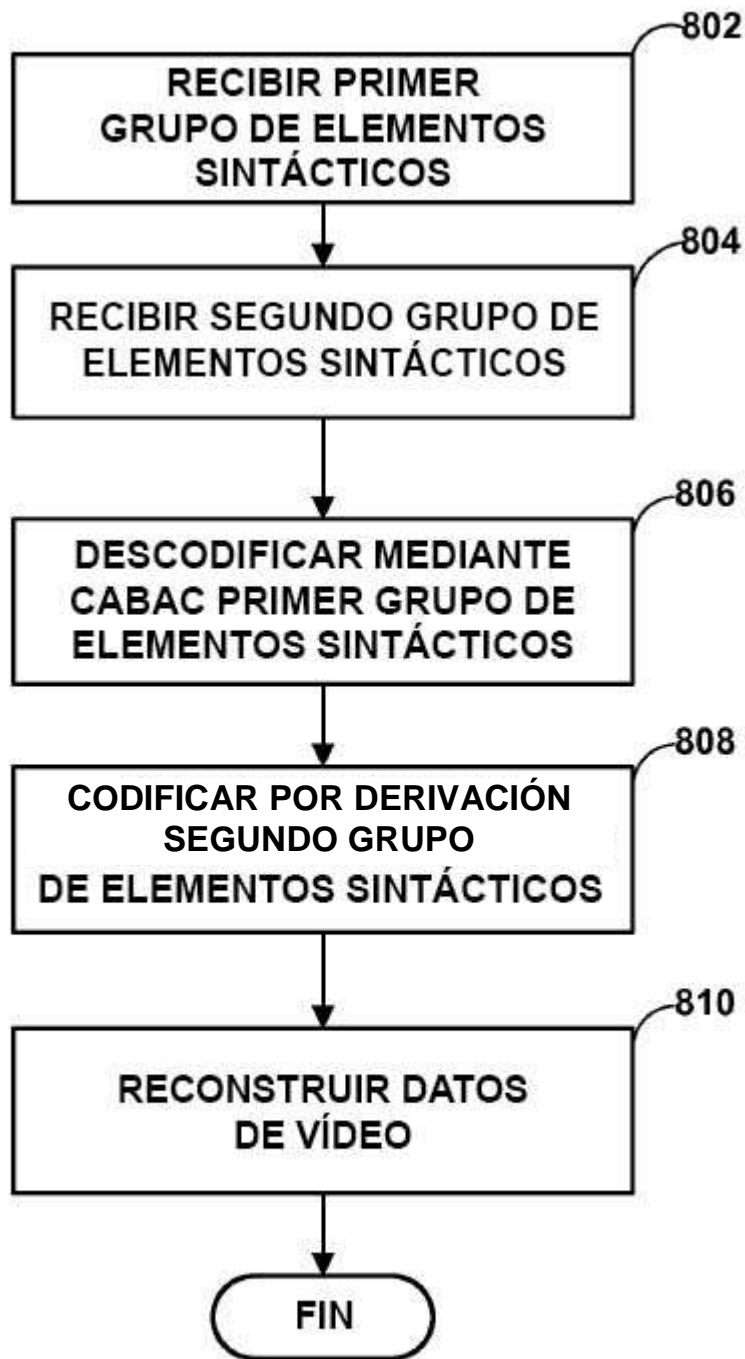


FIG. 8