

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 611 349**

51 Int. Cl.:

<b>H04N 19/176</b>	(2014.01)
<b>H04N 19/70</b>	(2014.01)
<b>H04N 19/96</b>	(2014.01)
<b>H04N 19/593</b>	(2014.01)
<b>H04N 19/11</b>	(2014.01)
<b>H04N 19/186</b>	(2014.01)
<b>H04N 19/645</b>	(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **27.09.2013 PCT/US2013/062141**
- 87 Fecha y número de publicación internacional: **10.04.2014 WO14055344**
- 96 Fecha de presentación y número de la solicitud europea: **27.09.2013 E 13776887 (5)**
- 97 Fecha y número de publicación de la concesión europea: **19.10.2016 EP 2904790**

54 Título: **Codificación de Indicador de bloque codificado (CBF) para el formato de muestra 4:2:2 en codificación de vídeo**

30 Prioridad:

**01.10.2012 US 201261708582 P**  
**26.09.2013 US 201314038524**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:  
**08.05.2017**

73 Titular/es:

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

72 Inventor/es:

**JOSHI, RAJAN LAXMAN;**  
**GUO, LIWEI y**  
**KARCZEWICZ, MARTA**

74 Agente/Representante:

**FORTEA LAGUNA, Juan José**

**ES 2 611 349 T3**

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

## DESCRIPCIÓN

Codificación de Indicador de bloque codificado (CBF) para el formato de muestra 4:2:2 en codificación de vídeo

- 5 Esta solicitud reivindica la ventaja de la solicitud provisional estadounidense N° 61/708.582, presentada el 1 de octubre de 2012.

### CAMPO TÉCNICO

- 10 Esta divulgación se refiere a la codificación de vídeo y, más en particular, a técnicas relacionadas con la división de datos residuales.

### ANTECEDENTES

- 15 Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de difusión directa digital, sistemas de difusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados teléfonos "inteligentes", dispositivos de videoconferencia, dispositivos de transmisión por flujo de vídeo, y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC) (H.264/AVC), la norma de Codificación de Vídeo de Alta Eficacia (HEVC) y las extensiones de tales normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital más eficazmente, implementando tales técnicas de compresión de vídeo.

- 30 Las técnicas de compresión de vídeo llevan a cabo la predicción espacial (intra-imagen) y/o la predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse bloques arbolados, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intra-codificado (I) de una imagen son codificados usando la predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen. Los bloques de vídeo de un fragmento inter-codificado (P o B) de una imagen pueden codificarse utilizando la predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen, o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

- 40 La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque inter-codificado se codifica según un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intra-codificado se codifica según una modalidad de intra-codificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de píxeles a un dominio de transformación, dando como resultado coeficientes de transformación residuales, los cuales pueden cuantizarse posteriormente. Los coeficientes de transformación cuantizados, inicialmente dispuestos en una formación bidimensional, pueden escanearse con el fin de producir un vector unidimensional de coeficientes de transformación, y puede aplicarse codificación por entropía a los coeficientes de transformación para lograr más compresión.

- 50 El artículo de Silcock P et al: "AHG12: Extensión de HM7 para dar soporte a formatos de crominancia adicionales", 10. Conferencia de JCT-VC, 101. Conferencia de MPEG; 11-7-2012 – 20-7-2012; Estocolmo; (Equipo de Colaboración Conjunta en Codificación de Vídeo de ISO/IEC JTC/SC29/WG11 e ITU-T SG. 16); URL:HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/; no. JCTVC-J0191, 2 de julio de 2012 (2012-07-02), XP030112553, describe un modelo basado en HM7.0 que proporciona soporte para formatos de crominancia 4:2:2, 4:4:4 y 4:0:0. En el modelo, se presta soporte a la codificación 4:2:0 y se informa de archivos de salida para 4:2:0 que coinciden con los proporcionados por el HM7.0 para las 8 configuraciones de prueba estándar y otras configuraciones no estándar, con tiempos de codificación / decodificación similares al HM7.0. También describe los indicadores de bloque codificados (CBF). En el HM7.0, se utilizan los CBF para indicar la presencia de coeficientes distintos de cero en la estructura de TU en árbol cuádruple. Estos indicadores se gestionan de forma diferente para luminancia y crominancia. Para luminancia, un CBF es codificado solo para las TU de hoja en el nivel más bajo del árbol cuádruple. Para crominancia, se codifica un CBF en cada nivel del árbol cuádruple.

### SUMARIO

- 65 La invención está definida en las reivindicaciones adjuntas, a las cuales se debería hacer referencia ahora. Esta divulgación se refiere a las técnicas para la división de unidades de transformación para los componentes de crominancia en un proceso de codificación de vídeo que utiliza un formato de muestra de color, tal como 4:2:2, que

produce bloques de crominancia rectangulares. Las técnicas pueden incluir dividir bloques de crominancia rectangulares en sub-bloques cuadrados, permitiendo la utilización de transformaciones cuadradas para los datos residuales en los bloques de crominancia. En algunos ejemplos, las técnicas pueden incluir el procesamiento de indicadores de bloque codificados (CBF) respectivos para los sub-bloques cuadrados que indican la presencia o ausencia de coeficientes de transformación distintos de cero en los sub-bloques. Las técnicas también, o alternativamente, pueden incluir, en algunos ejemplos, muestras de intra-codificación correspondientes a un sub-bloque cuadrado, por ejemplo, un sub-bloque cuadrado inferior, de un bloque de crominancia rectangular, sobre la base de muestras reconstruidas obtenidas de la intra-codificación del otro sub-bloque cuadrado, por ejemplo, un sub-bloque cuadrado superior.

En un ejemplo, la divulgación describe un procedimiento de decodificación de datos de vídeo, incluyendo el procedimiento la obtención de un bloque de crominancia rectangular que tenga sub-bloques cuadrados primero y segundo, la decodificación de un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, la decodificación de un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y la no decodificación de un CBF para el bloque de crominancia rectangular.

En un ejemplo, la divulgación describe un procedimiento de decodificación de datos de vídeo, incluyendo el procedimiento la división de una hoja de bloque de crominancia rectangular en sub-bloques cuadrados primero y segundo, la codificación de un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, la codificación de un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y la no codificación de un CBF para el bloque de crominancia rectangular.

En otro ejemplo, la divulgación describe un aparato para la decodificación de datos de vídeo que incluye uno o más procesadores configurados para obtener un bloque de crominancia rectangular que tiene sub-bloques cuadrados primero y segundo, decodificar un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, decodificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y no decodificar un CBF para el bloque de crominancia rectangular.

En otro ejemplo, la divulgación describe un aparato para la codificación de datos de vídeo, incluyendo uno o más procesadores configurados para dividir un bloque de crominancia rectangular en sub-bloques cuadrados primero y segundo, codificar un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, codificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y no codificar un CBF para el bloque de crominancia rectangular.

En otro ejemplo, la divulgación describe un aparato para la codificación de datos de vídeo que incluye medios para la obtención de un bloque de crominancia rectangular que tiene sub-bloques cuadrados primero y segundo, medios para la codificación de un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y medios para la codificación de un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, en el que ninguno, entre los medios para la codificación del primer CBF y los medios para la codificación del segundo CBF, codifica un CBF para el bloque de crominancia rectangular.

En otro ejemplo, la divulgación describe un medio de almacenamiento legible por ordenador. Teniendo el medio de almacenamiento legible por ordenador almacenado en el mismo instrucciones que, durante la ejecución, hacen que uno o más procesadores obtengan un bloque de crominancia rectangular que tiene sub-bloques cuadrados primero y segundo, codifiquen un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, codifiquen un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y no codifiquen un CBF para el bloque de crominancia rectangular.

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

**BREVE DESCRIPCIÓN DE LOS DIBUJOS**

La FIG. 1 es un diagrama de bloques que ilustra un sistema ejemplar de codificación y decodificación de vídeo que puede utilizar las técnicas descritas en esta divulgación.

5 La FIG. 2 es un diagrama de bloques que ilustra un codificador ejemplar de vídeo que puede implementar las técnicas descritas en esta divulgación.

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

10 Las FIGs. 4A a 4C son diagramas conceptuales que ilustran diferentes formatos de muestra de color para los componentes de luminancia y crominancia de una unidad de codificación.

15 La FIG. 5 es un diagrama conceptual que ilustra una unidad de codificación de tamaño 16x16, formateada según un formato de muestra de color 4:2:2.

La FIG. 6 es un diagrama conceptual que ilustra una estructura de descomposición en árbol cuádruple de una unidad de transformación.

20 La FIG. 7 es un diagrama conceptual que ilustra los niveles de descomposición en árbol cuádruple.

Las FIGs. 8A a 8B son diagramas conceptuales que ilustran técnicas para la división de unidades de transformación para un bloque de vídeo.

25 La FIG. 9 es un diagrama de flujo que ilustra un procedimiento ejemplar para la decodificación de datos de vídeo según los sistemas y procedimientos descritos en este documento.

La FIG. 10 es un diagrama de flujo que ilustra otro procedimiento ejemplar para la codificación de datos de vídeo según los sistemas y procedimientos descritos en este documento.

30 La FIG. 11 es un diagrama de flujo que ilustra un procedimiento ejemplar para la decodificación de datos de vídeo según los sistemas y procedimientos descritos en este documento.

35 La FIG. 12 es un diagrama de flujo que ilustra un procedimiento ejemplar para la codificación de datos de vídeo según los sistemas y procedimientos descritos en este documento.

## DESCRIPCIÓN DETALLADA

40 Según algunas normas de codificación de vídeo, una unidad de codificación (CU), incluyendo su componente de luminancia y sus componentes de crominancia, puede compartir la misma estructura de descomposición en árbol cuádruple para la división de la unidad de transformación (TU). Bajo diferentes formatos de toma de muestras de color, los correspondientes bloques de componentes de crominancia pueden tener diferentes tamaños y formas que el bloque de luminancia. Una versión final de la norma de codificación de vídeo de alta eficacia (HEVC) puede utilizar el formato 4:2:0. La versión final de la norma HEVC, cuando se combina con un ejemplo de extensiones de rango de HEVC consideradas, puede incluir otros formatos, tales como 4:2:2, por ejemplo. Para las extensiones de rango de HEVC consideradas, cuando se utiliza formato 4:2:2, por ejemplo, no todos los tamaños y formas diferentes de transformaciones que pueden obtenerse a partir de la división de las TU están definidos y disponibles para su uso por los bloques de componentes de crominancia. Para permitir el uso de transformaciones cuadradas, los bloques rectangulares de coeficientes de transformación para un componente de crominancia se pueden dividir en sub-bloques cuadrados, tal como se describe en esta divulgación. Por ejemplo, un bloque de crominancia rectangular puede tener un sub-bloque cuadrado superior y un sub-bloque cuadrado inferior.

55 En algunos ejemplos, la presente solicitud está relacionada con la codificación que utiliza formatos de color tales como el sub-muestreo de color 4:2:2 y 4:4:4. Por ejemplo, un ejemplo provee el uso de transformaciones cuadradas para una modalidad de muestreo de color 4:2:2 en la HEVC.

60 Un borrador reciente de la norma HEVC, denominado "Borrador 10 de trabajo de HEVC" o "WD10", se describe en el documento JCTVC-L1003v34, de Bross et al., titulado "Memoria descriptiva textual de la Codificación de Vídeo de Alta Eficacia (HEVC), Borrador 10 (para Norma Internacional de Borrador Final y Última Llamada)", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 12ª conferencia: Ginebra, Suiza, 14 a 23 de enero, 2013, que, a partir del jueves 6 de junio de 2013, puede descargarse desde: [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/12\\_Geneva/wg11/JCTVC-L1003-v34.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip).

65 Otro borrador (WD) de trabajo reciente de la norma HEVC, denominado "Borrador 6 de trabajo de la HEVC" o "WD6", se describe en el documento JCTVC-H1003, de Bross y col., titulado "Memoria Descriptiva Textual de la Codificación de Vídeo de Alta Eficacia (HEVC), Borrador 6", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 8ª conferencia: San Jose, California, EE. UU.,

febrero de 2012, que a partir del 24 de septiembre de 2012, puede descargarse desde: [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/8\\_San%20Jose/wg11/JCTVC-H1003-v22.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip).

Otro borrador (WD) de trabajo reciente de la norma HEVC, denominado "Borrador 8 de trabajo de la HEVC" o "WD8", se describe en el documento JCTVC-J1003, de Bross y col., titulado "Memoria Descriptiva Textual de la Codificación de Vídeo de Alta Eficacia (HEVC), Borrador 6", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 10ª conferencia: Estocolmo, Suecia, 11 a 20 de julio de 2012, que, a partir del 24 de septiembre de 2012, puede descargarse desde: [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/10\\_Stockholm/wg11/JCTVC-J1003-v8.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/10_Stockholm/wg11/JCTVC-J1003-v8.zip).

Las extensiones de rango de la HEVC se describen en el documento JCTVC-N1005\_v3, de Flynn et al, "Memoria descriptiva textual de las extensiones del rango de la codificación de vídeo de alta eficacia (HEVC): Borrador 4", Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 13ª conferencia: Incheon, COREA, 18 a 26 de abril de 2013, que, a partir del 22 de septiembre de 2013, puede descargarse desde: [http://phenix.it-sudparis.eu/jct/doc\\_end\\_user/current\\_document.php?id=8139](http://phenix.it-sudparis.eu/jct/doc_end_user/current_document.php?id=8139).

El JCT-VC está considerando un nuevo perfil para los formatos de color 4:4:2 y 4:4:4 como parte de las extensiones de rango de la HEVC. Para el formato 4:2:2, los componentes de crominancia se muestrean reducidos en un factor de 2 en la dirección horizontal, en comparación con el componente de luminancia. No hay muestreo reducido en la dirección vertical.

Según la HEVC, las tramas se pueden dividir en unidades de codificación máximas (LCU), que pueden comprender, por ejemplo, bloques de 32x32 píxeles o bloques de 64x64 píxeles, o bloques de otros tamaños. En general, las LCU pueden dividirse en unidades de sub-codificación (sub-CU), cada una de las cuales puede dividirse más. En general, el término unidad de codificación (CU) puede referirse a una LCU o cualquier sub-CU de la misma (por ejemplo, una sub-CU de una LCU o una sub-CU de otra sub-CU). Las CU pueden dividirse en cuatro bloques cuadrados, que no se superponen. Una LCU puede corresponderse con una estructura de datos de árbol cuádruple, que incluye uno o más nodos, donde un nodo raíz del árbol cuádruple corresponde a la propia LCU y otros nodos corresponden a las sub-CU de la LCU. Una CU sin particiones generalmente corresponde a un nodo hoja del árbol cuádruple (es decir, un nodo del árbol cuádruple que no tiene ningún nodo descendiente). Por consiguiente, una CU sin particiones puede denominarse una CU de nodo hoja. Las CU de nodo hoja generalmente incluyen una o más unidades de predicción (PU) que describen cómo se predicen los datos para la CU, y una o más unidades de transformación (TU) que corresponden a los datos residuales, es decir, diferencias píxel por píxel entre los datos predichos para la CU y los datos originales para la CU.

Un nodo hoja para una CU sin particiones puede incluir información que indica las modalidades de predicción para las PU de la CU. En algunos ejemplos, puede proporcionarse un árbol cuádruple residual (RQT) (también denominado árbol cuádruple de unidad de transformación (TU)) para las CU de nodo de hoja. El árbol cuádruple residual también puede incluir un nodo raíz, y uno o más nodos de hoja. El RQT puede incluir información, tal como información que indica las modalidades de predicción y/u otra información de predicción para las TU que corresponden a una parte de la correspondiente CU de nodo de hoja. En cualquier caso, la información para una CU puede indicar modalidades de predicción para una o más partes de la CU.

La información de modalidad proporcionada en el RQT puede indicar un tipo particular de modalidad de intra-predicción. Por ejemplo, la información proporcionada en el RQT puede indicar si se utiliza una modalidad de predicción DC, o si se utiliza una modalidad de intra-predicción direccional. En la modalidad de predicción DC, pueden agruparse valores para píxeles contiguos de los bloques codificados previamente, y utilizarse para formar valores predichos para los píxeles de la CU actual. En las modalidades de predicción de dirección, los valores de los píxeles contiguos de los bloques previamente codificados se pueden aplicar en una dirección particular para formar valores predichos para los píxeles de la CU actual.

En general, el diseño de codificación de coeficientes de la transformación de la HEVC no da soporte al caso en el que todos los coeficientes de transformación de una unidad de transformación (TU) son cero. En consecuencia, en la HEVC, normalmente un indicador de bloque codificado (CBF) se utiliza para señalar la presencia o ausencia de coeficientes distintos de cero en una TU. Algunos ejemplos de esta divulgación se refieren a enviar y/o recibir, o codificar y/o decodificar, un CBF respectivo para cada uno de dos sub-bloques de crominancia cuadrados de un bloque de crominancia rectangular, directamente, por ejemplo, sin enviar un indicador CBF para el bloque de crominancia rectangular.

Por ejemplo, los datos de vídeo pueden codificarse mediante la división de un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, en sub-bloques cuadrados primero y segundo, o mediante la obtención de un bloque de crominancia rectangular que incluye sub-bloques cuadrados primero y segundo. Los sub-bloques cuadrados primero y segundo pueden ser, por ejemplo, un sub-bloque cuadrado superior y un sub-bloque cuadrado inferior de un bloque rectangular. Un primer CBF puede ser codificado para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. Un segundo CBF, independiente del primer CBF, también puede ser codificado para el primer sub-bloque cuadrado,

para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero.

5 En algunos ejemplos, un CBF no es codificado generalmente para el bloque de crominancia rectangular, es decir, para la combinación de los sub-bloques cuadrados primero y segundo. De esta manera, cada sub-bloque cuadrado del bloque de crominancia rectangular tiene su propia indicación, es decir, su propio CBF, que se procesa para determinar si el respectivo sub-bloque incluye o no coeficientes de transformación distintos de cero. Por consiguiente, no hay necesidad de generar un CBF para el bloque general de crominancia rectangular.

10 En un ejemplo específico, si el tamaño de la CU de crominancia es de 8x16 y se subdivide en 4 unidades de transformación (TU) de tamaño 4x8, para cada componente de crominancia, por lo general, no se señalará un CBF al nivel del bloque de tamaño 4x8. En cambio, el bloque de crominancia de tamaño 4x8 (es decir, rectangular) puede subdividirse adicionalmente en dos sub-bloques de tamaño 4x4 (es decir, cuadrados). En este ejemplo, un CBF está codificado para cada bloque de tamaño 4x4, pero no para el bloque de tamaño 4x8.

15 El JCT-VC está considerando un nuevo perfil para formatos de color 4:2:2 y 4:4:4. Según algunas propuestas para el formato 4:2:2, los componentes de crominancia se muestrean reducidos en un factor de 2 en la dirección horizontal, en comparación con el componente de luminancia. No hay muestreo reducido en la dirección vertical.

20 Este tipo de sub-muestreo afecta a los tamaños de TU. Por ejemplo, considérese una CU de tamaño 16 (anchura) x 16 (altura). Téngase en cuenta que el árbol cuádruple residual subdivide la CU en 4 TU de tamaño 8x8 para la luminancia. Entonces, para los componentes de crominancia, el tamaño de las TU es de 4x8. Si los tamaños de transformación de luminancia máximo y mínimo son 32x32 y 4x4, respectivamente, entonces para componentes de crominancia de 4:2:2, tal vez se necesiten transformaciones de tamaños 16x32, 4x8 y 8x16. En el software de  
25 formato de crominancia extendido, pueden utilizarse transformaciones rectangulares correspondientes a estos tamaños. Esto afecta a la complejidad del hardware. En hardware, cada tamaño de transformación se implementa habitualmente como un bloque independiente. Por lo tanto, la adición de transformaciones rectangulares aumenta la complejidad del hardware. Además, el uso de transformaciones rectangulares de estos tamaños también requiere cambios en la cuantización (ajuste del QP en  $\pm 3$ ).

30 La división de un bloque de crominancia rectangular en sub-bloques cuadrados, tal como se ha descrito anteriormente, puede presentar algunas consideraciones para la intra-codificación de dichos sub-bloques. Las técnicas descritas en esta divulgación también, o alternativamente, pueden incluir, en algunos ejemplos, muestras de intra-codificación correspondientes a un sub-bloque cuadrado, por ejemplo, un sub-bloque cuadrado inferior, de un bloque de crominancia rectangular, sobre la base de muestras reconstruidas obtenidas a partir de la intra-codificación del otro sub-bloque cuadrado, por ejemplo, un sub-bloque cuadrado superior del bloque de crominancia rectangular. En un ejemplo, un procedimiento de codificación de datos de vídeo puede incluir la recepción de un bloque de crominancia rectangular que incluye un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. En algunos ejemplos, el bloque de crominancia rectangular puede ser un bloque de crominancia rectangular que no se transforma, es decir, un bloque de crominancia rectangular no transformado. En otros ejemplos, el bloque de crominancia rectangular puede ser un bloque de crominancia rectangular que se transforma, es decir, un bloque de transformación de crominancia rectangular. De manera similar, en algunos ejemplos, un sub-bloque cuadrado puede ser un sub-bloque cuadrado que no se transforma, es decir, un sub-bloque cuadrado no transformado. En otros ejemplos, un sub-bloque cuadrado puede ser un sub-bloque cuadrado que se transforma, es decir, un sub-bloque cuadrado de transformación. Por consiguiente, un bloque de crominancia rectangular puede ser un bloque de crominancia rectangular que se transforma o un bloque de crominancia rectangular que no se transforma. De forma similar, un sub-bloque cuadrado puede ser un sub-bloque cuadrado que se transforma o un sub-bloque cuadrado que no se transforma. Además, otros bloques pueden o no transformarse. En consecuencia, en algunos casos, una transformación se puede omitir para varios bloques.

50 Un primer bloque de datos de crominancia se puede reconstruir usando un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. Un segundo bloque de datos de crominancia intra-predichos puede generarse para el segundo sub-bloque cuadrado a partir de muestras de referencia del primer bloque reconstruido de datos de crominancia. Un segundo bloque de datos de crominancia se puede reconstruir usando el segundo bloque de datos de crominancia intra-predichos y unos segundos datos residuales del segundo sub-bloque cuadrado.

60 En el caso de la codificación de RQT intra-predictiva de la HEVC, se intercalan los pasos 1) intra-predicción y 2) transformación - cuantización - descuantización - transformación inversa. En la divulgación actual, se ilustran dos ejemplos. En un ejemplo, un módulo de predicción en un codificador de vídeo o un codificador de vídeo puede llevar a cabo la intra-predicción sobre el bloque de crominancia rectangular para cada hoja rectangular en la estructura de RQT, por ejemplo, en un módulo de intra-predicción. Entonces, el codificador de vídeo puede dividir el residuo de intra-predicción en dos bloques cuadrados y aplicar una transformación cuadrada a cada bloque. En otro ejemplo, el codificador de vídeo puede intercalar el proceso de predicción y transformar como en el caso del perfil principal de la HEVC. En este caso, el bloque de la hoja rectangular se divide en dos bloques de crominancia cuadrados (superior e inferior).

Un codificador realiza la intra-predicción sobre el sub-bloque de crominancia cuadrado superior, para generar un bloque de intra-predicción, es decir, un bloque de datos de crominancia intra-predichos, y datos residuales de intra-predicción que representan una diferencia entre los píxeles del bloque de datos de crominancia intra-predichos y los píxeles del sub-bloque de crominancia cuadrado superior. Los datos residuales de intra-predicción se someten a transformación, cuantización, cuantización inversa y transformación inversa en el codificador. Después de la cuantización inversa y la transformación inversa para reproducir los datos residuales, un sumador puede añadir los datos residuales resultantes nuevamente al bloque de intra-predicción para formar un sub-bloque cuadrado superior reconstruido de datos de crominancia. A continuación, las muestras reconstruidas a partir del sub-bloque cuadrado superior (y/u otras muestras reconstruidas) se utilizan como muestras de intra-predicción para realizar la intra-predicción para el sub-bloque cuadrado inferior. Por ejemplo, un codificador puede generar un bloque de datos de crominancia intra-predichos para el sub-bloque de crominancia cuadrado inferior utilizando muestras de referencia del sub-bloque de crominancia cuadrado superior reconstruido. A continuación, en un ejemplo, el codificador de vídeo genera datos residuales para el sub-bloque de crominancia cuadrado inferior, que representan una diferencia entre los datos del sub-bloque de crominancia cuadrado inferior y el bloque de datos de crominancia intra-predichos, generados para el sub-bloque cuadrado inferior. El codificador realiza la transformación, cuantización y codificación por entropía de los datos residuales para el sub-bloque cuadrado inferior, por ejemplo, para su uso por un decodificador. Un decodificador de vídeo decodifica el sub-bloque cuadrado superior, y utiliza los datos de píxeles reconstruidos, provenientes del sub-bloque cuadrado superior, como muestras de referencia de intra-predicción para generar el bloque intra-predicho para el sub-bloque cuadrado inferior en el lado del decodificador, obtiene datos residuales para el sub-bloque cuadrado inferior, y luego reconstruye el sub-bloque cuadrado inferior mediante la adición del bloque intra-predicho y los datos residuales.

Por lo tanto, en este ejemplo de intra-codificación, un codificador de vídeo puede dividir un bloque de crominancia rectangular en sub-bloques cuadrados superior e inferior. El codificador de vídeo puede seleccionar una modalidad de intra-codificación para el sub-bloque cuadrado superior, y aplicar la modalidad de intra-codificación para generar un bloque de datos de crominancia intra-predichos para el sub-bloque cuadrado superior. El bloque intra-predicho de datos de crominancia se puede generar, por ejemplo, utilizando muestras de referencia de intra-predicción a partir de uno o más bloques que sean espacialmente adyacentes al sub-bloque cuadrado superior, por ejemplo, ya sea como un bloque a la izquierda o un bloque superior. El codificador de vídeo genera un bloque de datos residuales en base a una diferencia entre los datos de crominancia originales del sub-bloque cuadrado superior y el bloque intra-predicho de datos de crominancia. El codificador de vídeo transforma el bloque de datos residuales para el sub-bloque cuadrado superior en coeficientes de transformación y cuantiza los coeficientes de transformación que representan los datos residuales. El codificador de vídeo codifica por entropía la información de modalidad de intra-codificación para el sub-bloque cuadrado superior, para permitir la generación del bloque intra-predicho de datos de crominancia por el decodificador y codifica por entropía los coeficientes de transformación cuantizados para permitir al decodificador obtener los datos residuales.

En el lado del codificador, para intra-codificar el sub-bloque cuadrado inferior utilizando muestras de referencia de intra-predicción, provenientes del sub-bloque cuadrado superior, el codificador de vídeo cuantiza inversamente y transforma inversamente los coeficientes de transformación para obtener los datos residuales para el sub-bloque cuadrado superior de datos de crominancia, genera el bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado superior de datos de crominancia y, a continuación, reconstruye el bloque cuadrado superior de datos de crominancia mediante la adición de los datos residuales al bloque intra-predicho de datos de crominancia. Algunos de los datos de crominancia provenientes del sub-bloque cuadrado superior reconstruido pueden a continuación utilizarse como muestras de referencia de intra-predicción para intra-codificar el sub-bloque cuadrado inferior de datos de crominancia.

Por ejemplo, el codificador puede seleccionar una modalidad de intra-codificación para el sub-bloque cuadrado inferior, y aplicar la modalidad de intra-codificación para generar un bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado inferior. El bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado inferior se puede generar, por ejemplo, utilizando muestras de referencia de intra-predicción reconstruidas a partir del sub-bloque cuadrado superior, y/o usando muestras de referencia de intra-predicción de otro bloque espacialmente adyacente, tal como un bloque adyacente izquierdo. Si se utilizan muestras de intra-predicción provenientes del sub-bloque cuadrado superior, el codificador de vídeo genera un bloque intra-predictivo de datos de crominancia para el sub-bloque cuadrado inferior en base a muestras de intra-predicción del sub-bloque cuadrado superior.

El codificador de vídeo puede generar un bloque de datos residuales en base a una diferencia entre los datos de crominancia originales del sub-bloque cuadrado inferior y el bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado inferior. El codificador de vídeo puede transformar el bloque de datos residuales para el sub-bloque cuadrado inferior en coeficientes de transformación y cuantizar los coeficientes de transformación que representan los datos residuales. El codificador de vídeo puede codificar por entropía la información de modalidad de intra-codificación para el sub-bloque cuadrado inferior, para permitir la generación del bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado inferior mediante el decodificador, y codifica por entropía los coeficientes de transformación cuantizados para permitir que el decodificador obtenga los datos residuales para el

sub-bloque cuadrado inferior.

En el lado del decodificador, un decodificador recibe el flujo de bits codificado y obtiene un bloque de crominancia rectangular que se divide en un sub-bloque cuadrado superior y un sub-bloque cuadrado inferior. El decodificador  
 5 intra-decodifica el sub-bloque cuadrado superior. Por ejemplo, el decodificador genera un bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado superior, utilizando muestras de referencias de uno o más bloques espacialmente adyacentes, cuantiza inversamente y transforma inversamente los coeficientes de transformación para el sub-bloque cuadrado superior, para producir datos residuales para el sub-bloque cuadrado superior, y luego reconstruye el sub-bloque cuadrado superior de datos de crominancia sumando el bloque intra-predicho de datos de crominancia y los datos residuales. A continuación, el sub-bloque cuadrado superior reconstruido puede ser utilizado para la intra-predicción del sub-bloque cuadrado inferior.

Por ejemplo, el decodificador de vídeo puede generar un bloque intra-predicho de datos de crominancia para el sub-bloque cuadrado inferior, utilizando muestras de referencia de intra-predicción del sub-bloque cuadrado superior reconstruido de datos de crominancia. El decodificador de vídeo cuantiza inversamente y transforma inversamente los coeficientes de transformación para el sub-bloque cuadrado inferior, para producir datos residuales para el sub-bloque cuadrado inferior, y a continuación reconstruye el sub-bloque cuadrado inferior de datos de crominancia sumando el bloque intra-predicho de datos de crominancia y los datos residuales para el sub-bloque cuadrado inferior. En algunos ejemplos, alternativamente o adicionalmente, el sub-bloque cuadrado inferior puede intra-predicirse usando muestras de referencia de otros bloques espacialmente adyacentes, tales como un bloque contiguo izquierdo, ya sea solo o en combinación con muestras de referencia del sub-bloque cuadrado superior. En la forma descrita anteriormente, un sub-bloque cuadrado superior se puede utilizar para intra-predicir un sub-bloque cuadrado inferior de un bloque de crominancia rectangular.

A continuación se describirán varios aspectos de un proceso de codificación de vídeo. En general, los dispositivos de vídeo digital implementan técnicas de compresión de vídeo para codificar y decodificar de manera más eficaz la información de vídeo digital. Las técnicas de compresión de vídeo pueden definirse según una norma de codificación de vídeo. Los ejemplos de 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 extensiones de codificación de vídeo ajustable a escala (SVC) y de codificación de vídeo de múltiples vistas (MVC). Además, hay una nueva norma de codificación de vídeo, concretamente, la codificación de vídeo de alta eficacia (HEVC), que está siendo desarrollada por el Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC.

Los esfuerzos de la normalización de la HEVC se basan en un modelo de un dispositivo de codificación de vídeo denominado modelo de prueba de la HEVC (HM). El HM supone mejoras en las capacidades de los dispositivos actuales de codificación de vídeo con respecto a los dispositivos de codificación de vídeo disponibles durante el desarrollo de otras normas anteriores de codificación de vídeo, por ejemplo, la ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modalidades de codificación de intra-predicción, la norma HEVC proporciona hasta treinta y cinco modalidades de codificación de intra-predicción.

La norma HEVC también puede denominarse ISO/IEC 23008-HEVC, que se pretende que sea el número estándar para la versión publicada de la HEVC. La codificación, según algunos de los aspectos propuestos en la actualidad de la norma HEVC en desarrollo, se describirá en esta solicitud con fines ilustrativos. Sin embargo, las técnicas descritas en esta divulgación también pueden ser útiles para, y aplicarse a, otros procesos de codificación de vídeo, tales como los que se definen según la ITU-T H.264 u otros procesos de codificación de vídeo, estándar o patentados.

Un codificador de vídeo típico que funciona según una norma de codificación de vídeo, como la HEVC WD10, divide cada trama (es decir, imagen) de una secuencia de vídeo original en regiones rectangulares contiguas llamadas "bloques" o "unidades de codificación". Estos bloques pueden codificarse aplicando técnicas de predicción espacial (intra-trama) y/o temporal (inter-trama) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Una predicción espacial puede denominarse una "intra-modalidad" (Modalidad I), y una predicción temporal puede denominarse una "inter-modalidad" (modalidad P o modalidad B). Las técnicas de predicción generan un bloque predictivo de datos de vídeo. Un bloque de datos de vídeo originales a codificar se compara con el bloque predictivo.

La diferencia entre los valores de píxeles del bloque original de datos de vídeo y los valores de píxel del bloque predictivo puede denominarse datos residuales. Los datos residuales pueden incluir los datos residuales para el componente de luminancia y los componentes de crominancia. Los datos residuales son habitualmente una formación de la diferencia entre los valores de píxel de un bloque predictivo y el bloque original de datos de vídeo. Una transformación, por ejemplo, una transformación de coseno discreta (DCT), una transformación entera, una transformación de ondículas, o una transformación conceptualmente similar, puede aplicarse a los datos residuales durante el proceso de codificación para generar un conjunto correspondiente de coeficientes de transformación. Por lo tanto, el bloque original de vídeo se puede reconstruir mediante la realización de una transformación inversa de los coeficientes de transformación y la adición de los datos residuales al bloque predictivo. Los coeficientes de

transformación también pueden cuantizarse. Es decir, los valores de los coeficientes de transformación pueden representarse como una cadena de bits según una profundidad de bits definida. En algunos casos, la cuantización puede dar como resultado la representación de coeficientes de transformación de bajo valor como cero. Los coeficientes de transformación cuantizados pueden ser denominados niveles de coeficiente de transformación.

5 Para la codificación de vídeo según la HEVC, como ejemplo, una trama de vídeo puede dividirse en unidades de codificación. Una unidad de codificación (CU) generalmente se refiere a una región de imagen rectangular que sirve como una unidad básica sobre la cual se aplican diversas herramientas de codificación para la compresión de vídeo. Una CU es habitualmente cuadrada, y puede considerarse similar a un llamado "macro-bloque", descrito en otras  
10 normas de codificación de vídeo, tales como, por ejemplo, la ITU-T H.264. Una CU se puede considerar como una formación de valores de muestras de vídeo. Los valores de muestras de vídeo también pueden denominarse elementos de imagen, píxeles o pels. El tamaño de una CU se puede definir según un número determinado de muestras horizontales y verticales. Por lo tanto, una CU puede describirse como una CU de tamaño NxN o NxM.

15 En esta divulgación, "NxN" y "N por N" pueden usarse de manera intercambiable para hacer referencia a las dimensiones de píxeles de un bloque de vídeo, en lo que respecta a dimensiones verticales y horizontales, por ejemplo, 16x16 píxeles o 16 por 16 píxeles. En general, un bloque de tamaño 16 x 16 tendrá 16 píxeles en la dirección vertical (y = 16) y 16 píxeles en la dirección horizontal (x = 16). Asimismo, un bloque de tamaño NxN tiene generalmente N píxeles en la dirección vertical y N píxeles en la dirección horizontal, donde N representa un valor  
20 entero no negativo. Los píxeles en un bloque pueden estar dispuestos 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 N x M píxeles, donde M no es necesariamente igual a N.

25 Para lograr una mejor eficacia de codificación, una CU puede tener tamaños variables en función del contenido de vídeo. Una CU por lo general tiene un componente de luminancia, denominado Y, y dos componentes de crominancia, denominados U y V. Los dos componentes de crominancia U y V también pueden denominarse C<sub>b</sub> y C<sub>r</sub>, respectivamente. Además, según la HEVC, los datos sintácticos dentro de un flujo de bits pueden definir una unidad de codificación máxima (LCU), que es la CU más grande para una trama o imagen en términos del número de muestras. El tamaño de una CU se define generalmente según el número de muestras de luminancia horizontales y  
30 verticales. Por lo general, una LCU incluye 64x64 muestras de luminancia. Las CU de otras dimensiones se puede generar mediante la división recursiva de una LCU en las sub-CU. Los datos sintácticos para un flujo de bits pueden definir un número máximo de veces en que puede dividirse una LCU, lo que se denomina profundidad de CU. Por consiguiente, un flujo de bits también puede definir una mínima unidad de codificación (SCU). Por lo general, una SCU incluye 8x8 muestras de luminancia. De este modo, en un ejemplo, se pueden generar cuatro CU de tamaño  
35 32x32 mediante la división de una LCU de tamaño 64x64 en cuatro sub-CU y cada una de las CU de tamaño 32x32 puede dividirse además en dieciséis CU de tamaño 8x8 o, en otro ejemplo, un bloque de tamaño 32x32 (FIG. 1) puede dividirse en cuatro bloques de tamaño 16x16 (FIG. 2).

40 Un formato de muestreo de vídeo, que también se puede denominar un formato de crominancia, o formato de muestra de color, puede definir el número de muestras de crominancia incluidas en una CU con respecto al número de muestras de luminancia incluidas en un CU. En función del formato de muestreo de vídeo para los componentes de crominancia, el tamaño, en términos del número de muestras, de los componentes U y V puede ser el mismo que, o diferente a, el tamaño del componente Y. En las normas de codificación de vídeo H.264/AVC y HEVC WD10, se define un valor llamado `idc_formato_croma` para indicar diferentes formatos de muestreo de los componentes de  
45 crominancia, en relación con el componente de luminancia. La tabla 1 ilustra la relación entre los valores de `idc_formato_croma` y los formatos de crominancia asociados.

**Tabla 1: diferentes formatos de crominancia definidos en la norma H.264/AVC**

50

<code>idc_formato_croma</code>	formato de crominancia	SubAnchuraC	SubAlturaC
0	monocromo	-	-
1	4:2:0	2	2
2	4:2:2	2	1
3	4:4:4	1	1

55 En la Tabla 1, las variables `SubAnchuraC` y `SubAlturaC` se pueden utilizar para indicar la razón entre las frecuencias de muestreo horizontal y vertical, entre el número de muestras para el componente de luminancia y el número de muestras para cada componente de crominancia. En los formatos de crominancia descritos en la Tabla 1, los dos componentes de crominancia tienen la misma frecuencia de muestreo.

60 En el ejemplo de la Tabla 1, para el formato 4:2:0, la frecuencia de muestreo para el componente de luminancia es el doble de la de los componentes de crominancia para ambas direcciones horizontal y vertical. Como resultado, para una unidad de codificación formateada según el formato 4:2:0, la anchura y la altura de una formación de muestras para el componente de luminancia son el doble de las de cada formación de muestras para los componentes de

5 crominancia. De manera similar, para una unidad de codificación formateada según el formato 4:2:2, la anchura de una formación de muestras para el componente de luminancia es el doble de la anchura de una formación de muestras para cada componente de crominancia, pero la altura de la formación de muestras para el componente de luminancia es igual a la altura de una formación de muestras para cada componente de crominancia. Para una

10 La FIG. 1 es un diagrama de bloques que ilustra un sistema ejemplar de codificación y decodificación de vídeo 10 que puede utilizar las técnicas descritas en esta divulgación. En particular, el sistema de codificación de vídeo 10 puede configurarse para realizar las técnicas para la división de TU entre muestras de componentes de luminancia y crominancia cuando se utiliza el formato de muestra 4:2:2. Como se ilustra en la FIG. 3, el sistema 10 incluye un codificador de vídeo 20 que genera datos de vídeo codificados, a decodificar en un momento posterior mediante un decodificador de vídeo 30. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera

15 entre una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos telefónicos manuales tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de flujos de transmisión de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

20 El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a decodificar, mediante un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificado desde el codificador de vídeo 20 al decodificador de vídeo 30. En un ejemplo, el enlace 16 puede comprender un medio de comunicación para permitir al codificador de vídeo 20 transmitir datos de vídeo codificados directamente al decodificador de vídeo 30 en tiempo real. Los datos de vídeo codificados pueden ser modulados según una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitidos al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área

25 extensa o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

30 Como alternativa, los datos codificados pueden ser emitidos desde la interfaz de salida 22 a un dispositivo de almacenamiento 34. De manera similar, la interfaz de entrada 28 puede acceder a los datos codificados desde el dispositivo de almacenamiento 34. El dispositivo de almacenamiento 34 puede incluir cualquiera entre una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco, discos Blu-ray, DVD, CD-ROM, memoria flash, memoria volátil o no volátil, u otros medios adecuados cualesquiera de almacenamiento digital, para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de

35 almacenamiento 34 puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda retener los datos de vídeo codificado, generados por el codificador de vídeo 20. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados procedentes del dispositivo de almacenamiento 34, mediante flujos de transmisión o descarga mediante la interfaz de entrada 28. El servidor de ficheros puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al decodificador de vídeo 30. Los ejemplos de servidores de ficheros incluyen un servidor de la Red (por ejemplo, para una sede de la Red), un servidor del FTP, dispositivos de almacenamiento conectados a la red (NAS) o una unidad de disco local. El decodificador de vídeo 30 puede acceder a los datos de vídeo codificados mediante cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión por cable (por ejemplo, DSL, módem de cable, etc.), o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificado, almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificado desde el dispositivo de almacenamiento 34 puede ser una transmisión por flujo, una transmisión de descarga o una combinación de ambas.

40 Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, en soporte de cualquiera entre una diversidad de aplicaciones de multimedia, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo por flujo, por ejemplo, mediante Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, decodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10

45 puede configurarse para dar soporte a la transmisión de vídeo unidireccional o bidireccional, para prestar soporte a aplicaciones tales como los flujos de transmisión de vídeo, la reproducción de vídeo, la difusión de vídeo y/o la videotelefonía.

50 En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye un origen de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/demodulador (módem) y/o un transmisor. El origen de vídeo 18 puede incluir un origen tal como un dispositivo de captura de

vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente capturado, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de tales orígenes. Como un ejemplo, si el origen de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, las técnicas descritas en esta divulgación pueden aplicarse a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas.

El codificador de vídeo 20 codifica el vídeo capturado, pre-capturado o generado por ordenador, recibido desde el origen de vídeo 18. El vídeo capturado, pre-capturado o generado por ordenador puede formatearse según cualquiera de los formatos de muestra descritos anteriormente, incluyendo los formatos de muestra 4:2:0, 4:2:2 o 4:4:4. El codificador de vídeo 20 puede realizar la codificación de vídeo sobre vídeo formateado según cualquiera de los formatos de muestra 4:2:0, 4:2:2 o 4:4:4. En algunos casos, el codificador de vídeo 20 aumenta o reduce las muestras del vídeo capturado, pre-capturado, o generado por ordenador, como parte del proceso de codificación. Por ejemplo, el vídeo capturado puede formatearse según el formato de muestra 4:4:4, el codificador de vídeo 20 puede reducir las muestras del vídeo capturado al formato 4:2:2 y realizar la codificación de vídeo sobre el vídeo de muestras reducidas. Los datos de vídeo codificados pueden ser transmitidos directamente al dispositivo de destino 14 mediante la interfaz de salida 22 del dispositivo de origen 12. Los datos de vídeo codificados también (o como alternativa) pueden almacenarse en el dispositivo de almacenamiento 34 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para decodificación y/o reproducción.

El dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 recibe los datos de vídeo codificados por el enlace 16. Los datos de vídeo codificado, comunicados por el enlace 16, o proporcionados en el dispositivo de almacenamiento 34, pueden incluir una diversidad de elementos sintácticos generados por el codificador de vídeo 20, para su uso por un decodificador de vídeo, tal como el decodificador de vídeo 30, en la decodificación de los datos de vídeo. Tales elementos sintácticos pueden incluirse con los datos de vídeo codificados, transmitidos en un medio de comunicación, almacenarse en un medio de almacenamiento o almacenarse en un servidor de ficheros.

El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también configurarse para mantener interfaces con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 exhibe los datos de vídeo decodificado a un usuario, y puede comprender cualquiera entre una diversidad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar según una norma de compresión de vídeo, tal como la norma de Codificación de Vídeo de Alta Eficacia (HEVC), actualmente en desarrollo, y puede generalmente cumplir el Modelo de Prueba de la HEVC actual (HM) o un futuro HM.

Como alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar según otras normas patentadas o industriales, tales como la norma ITU-T H.264, también denominada MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), o revisiones o extensiones de tales normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen la MPEG-2 y la ITU-T H.263.

Aunque no se muestra en la FIG. 1, en algunos aspectos, tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden estar integrados con un codificador y decodificador de audio, y pueden incluir unidades adecuadas de multiplexado y demultiplexado, u otro hardware y software, para llevar a cabo la codificación, tanto de audio como de vídeo, en un flujo de datos común o en flujos de datos diferentes. Si procede, en algunos ejemplos, las unidades de multiplexado y demultiplexado pueden ajustarse al protocolo de multiplexado ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden implementarse como cualquiera entre una diversidad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables en el terreno (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas son implementadas parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio no transitorio adecuado, legible por ordenador, y ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden incluirse en uno o más codificadores o decodificadores, donde cualquiera de los mismos puede estar integrado como parte de un codificador/decodificador combinado (códec) en un dispositivo respectivo.

Como se ha descrito anteriormente, en la HEVC WD10, una trama o imagen de vídeo puede dividirse en una secuencia de bloques arbolados, o unidades máximas de codificación (LCU), que incluyen muestras tanto de luminancia como de crominancia. Las LCU pueden dividirse de forma recursiva en CU que incluyen las PU y TU asociadas. En algunos ejemplos, el tamaño de una CU puede variar desde muestras de 8x8 hasta el tamaño del bloque arbolado, con un máximo de muestras de tamaño 64x64, o más. Además, en la HEVC WD10, una trama o imagen de vídeo puede dividirse en uno o más fragmentos, en donde un fragmento incluye un cierto número de LCU consecutivas en orden de codificación. Una trama de vídeo puede dividirse en fragmentos basándose en modalidades de codificación asociadas a las CU incluidas en un fragmento. Por ejemplo, un fragmento puede definirse de forma que todas las CU incluidas en el fragmento incluyan una modalidad de codificación común, tal como: modalidad de omisión, modalidad directa, modalidad de intra-predicción o modalidad de inter-predicción.

Una serie de tramas o imágenes de vídeo puede codificarse como parte de una secuencia de vídeo codificado de grupos de imágenes. Un grupo de imágenes (GOP) comprende generalmente una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos, en una cabecera del GOP, en una cabecera de una o más de las imágenes, o en otras ubicaciones, que describen un cierto número de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen una modalidad de codificación para el fragmento respectivo. Como se describe en este documento, un codificador de vídeo 20 actúa habitualmente sobre bloques de vídeo dentro de los fragmentos de vídeo individuales, con el fin de codificar los datos de vídeo. Sin embargo, el codificador de vídeo 20 puede también realizar procesos de codificación de vídeo de nivel superior, tales como dividir una trama de vídeo en fragmentos y/o generar la sintaxis de nivel de GOP o nivel de fragmento.

Como se describió anteriormente, los datos sintácticos asociados a las CU pueden describir la división de una CU en una o más PU. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU según un árbol cuádruple. En algunos ejemplos, una TU puede tener forma cuadrada o no cuadrada. En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU está codificada en intra-modalidad, la PU puede incluir datos que describen una modalidad de intra-predicción para la PU. Como otro ejemplo, cuando la PU está codificada en inter-modalidad, la PU puede incluir datos que definen información de movimiento para la PU. Los datos que definen la información de movimiento para una PU pueden describir, por ejemplo, un vector de movimiento que tiene un componente horizontal, un componente vertical, una resolución (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una dirección de predicción que indica una lista de imágenes de referencia (por ejemplo, Lista 0, Lista 1 o Lista C) para el vector de movimiento, y un índice de imagen de referencia que indica una imagen de referencia hacia la cual apunta el vector de movimiento dentro de una lista de imágenes de referencia. Por lo tanto, un PU puede considerarse una unidad básica para el transporte de información relacionada con el proceso de predicción. En un ejemplo, el codificador de vídeo 20 puede estar configurado para codificar datos de vídeo según las estructuras de PU con soporte en el modelo de prueba de la HEVC (HM).

Como ejemplo, el HM da soporte a la predicción en diversos tamaños de PU. Suponiendo que el tamaño de una CU específica sea  $2N \times 2N$ , el HM presta soporte a la intra-predicción en tamaños de PU de  $2N \times 2N$  o  $N \times N$ , y a la intra-predicción en tamaños de PU simétricas de  $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$  o  $N \times N$ . El HM también presta soporte a la división asimétrica para la inter-predicción en tamaños de PU de  $2N \times nU$ ,  $2N \times nD$ ,  $nL \times 2N$  y  $nR \times 2N$ . En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida entre el 25 % y el 75 %. La parte de la CU correspondiente a la partición del 25 % está indicada por una "n" seguida por una indicación de "Arriba", "Abajo", "Izquierda" o "Derecha". Así, por ejemplo, "2NxnU" se refiere a una CU de tamaño  $2N \times 2N$  que está dividida horizontalmente, con una PU de tamaño  $2N \times 0,5N$  encima y una PU de tamaño  $2N \times 1,5N$  debajo.

Como se ha descrito anteriormente, una TU es la unidad básica que se utiliza para procesos de transformación y cuantización. Una CU dada con una o más PU también puede incluir una o más unidades de transformación (TU). Además, en la HEVC WD10, las TU pueden tener un tamaño basado en la partición de RQT. En otras palabras, la partición de RQT puede definir tamaños de transformación y puede especificarse para una CU y puede ser independiente de la partición de PU en algunos ejemplos, aunque esto puede no ser siempre el caso. En algunos ejemplos, las TU pueden tener el mismo tamaño o uno más pequeño que las PU. En algunos ejemplos, por ejemplo, para la inter-modalidad, las TU pueden ser incluso mayores que las PU. En tal caso, el tamaño en el que se realiza la predicción puede ser el mismo que el tamaño en el que se aplica la transformación. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas usando una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT). Los nodos de hoja del RQT pueden denominarse unidades de transformación (TU).

Tras la intra-predicción o la inter-predicción, el codificador de vídeo 20 puede calcular datos de vídeo residuales correspondientes a la PU. Los valores residuales comprenden valores de diferencia de píxeles que representan las diferencias entre los valores de los píxeles del bloque original a codificar y los valores de los píxeles del bloque predictivo formado por intra-predicción o inter-predicción. Los datos de vídeo residuales en la unidad de predicción, es decir, los valores residuales de diferencia de píxel, pueden transformarse en coeficientes de transformación según la estructura de TU para producir coeficientes de transformación. Por lo tanto, los datos residuales pueden transformarse desde el dominio de píxeles a un dominio de transformación. En algunos casos (por ejemplo,

modalidad de omisión de transformación), la transformación puede omitirse.

Tras cualquier transformación para producir coeficientes de transformación, el codificador de vídeo 20 puede realizar la cuantización de los coeficientes de transformación. La cuantización se refiere generalmente a un proceso en el que los coeficientes de transformación se cuantizan para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando una compresión adicional. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos de, o todos, los coeficientes. Por ejemplo, un valor de  $n$  bits puede redondearse por lo bajo a un valor de  $m$  bits durante la cuantización, donde  $n$  es mayor que  $m$ .

En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de escaneo predefinido para escanear los coeficientes de transformación cuantizados, para producir un vector serializado que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar un escaneo adaptativo. Después de escanear los coeficientes de transformación cuantizados, para formar un vector unidimensional, el codificador de vídeo 20 puede codificar por entropía el vector unidimensional, por ejemplo, según la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto y de base sintáctica (SBAC), la codificación por Entropía por División de Intervalos de Probabilidad (PIPE) u otra metodología de codificación por entropía. El codificador de vídeo 20 también puede codificar por entropía elementos sintácticos asociados a los datos de vídeo codificados, para su uso por el decodificador de vídeo 30, en la decodificación de los datos de vídeo.

Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto dentro de un modelo contextual a un símbolo a transmitir. El contexto puede referirse, por ejemplo, a si los valores adyacentes del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC pueden ser construidas de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un poco de ahorro con respecto, por ejemplo, a usar palabras de código de igual longitud para cada símbolo a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.

Como se ha descrito anteriormente, la codificación de coeficientes de transformación de la HEVC no da soporte al caso en el que todos los coeficientes son cero. La HEVC no está diseñada para analizar sintácticamente los bloques que tienen solo coeficientes con valor cero. En consecuencia, la HEVC puede necesitar ser capaz de omitir los bloques que tienen solo coeficientes de valor cero. Esto se puede lograr mediante el uso del indicador de bloque codificado (CBF) que normalmente se utiliza para señalar la presencia o ausencia de coeficientes distintos de cero. En algunos ejemplos de esta divulgación, el codificador de vídeo 20 señala dos CBF para sub-bloques de crominancia, un CBF para cada uno de dos sub-bloques de crominancia cuadrados de coeficientes de transformación, de un bloque rectangular, directamente, sin enviar un indicador CBF para el bloque rectangular. A continuación, el decodificador de vídeo 30 puede recibir dos CBF, un CBF para cada uno de dos sub-bloques de crominancia cuadrados. El codificador de vídeo 20 no señala ningún CBF en el flujo de bits codificado (en este ejemplo) para el bloque rectangular y, por lo tanto, el decodificador de vídeo 30 no recibirá ningún CBF para el bloque de crominancia rectangular en el flujo de bits codificado. Más específicamente, el codificador de vídeo 20, dentro del dispositivo de origen 12, puede codificar dos CBF, un CBF para cada uno de dos bloques, directamente para cada bloque, sin codificar un indicador CBF para un bloque rectangular. El CBF para el bloque rectangular indicaría que al menos uno de los sub-bloques incluye coeficientes distintos de cero. En algunos ejemplos, con CBF individuales para cada uno de los sub-bloques, sin embargo, un CBF para el bloque rectangular no es necesario. En otros ejemplos, sin embargo, el CBF rectangular todavía puede enviarse. Si el CBF rectangular es cero, el envío de los CBF para los sub-bloques cuadrados puede omitirse. Los CBF pueden almacenarse en el dispositivo de almacenamiento 34 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para decodificación y/o reproducción. En otro ejemplo, estos CBF pueden codificarse en un flujo de bits que se transmite a través de la interfaz de salida 22, mediante el enlace 16 y/o registrarse en un almacenamiento de flujos de bits codificados en el dispositivo de almacenamiento 34. El decodificador de vídeo 30 en el dispositivo de destino 14 puede a continuación recibir los dos CBF desde el enlace 16 y/o el dispositivo de almacenamiento 34 en la interfaz de entrada 28. El decodificador de vídeo 30 puede a continuación decodificar los CBF.

Por ejemplo, los datos de vídeo pueden codificarse dividiendo un bloque de crominancia rectangular, por ejemplo, generado según un formato de muestra de color 4:2:2, por ejemplo, en una hoja de árbol cuádruple en sub-bloques cuadrados primero y segundo. Un CBF puede estar codificado para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. Otro CBF también puede estar codificado para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. En este ejemplo, un CBF no está codificado generalmente para el bloque de crominancia rectangular, es decir, la combinación de los sub-bloques cuadrados primero y segundo para el componente de crominancia.

En un ejemplo específico, si el tamaño de la CU de crominancia es de 8x16 y se subdivide en cuatro unidades de transformación de tamaño 4x8, para cada componente de crominancia, por lo general no se señala un CBF a nivel del bloque de tamaño 4x8. El bloque de crominancia rectangular de tamaño 4x8 puede subdividirse en dos sub-

bloques cuadrados de tamaño 4x4. Un CBF está codificado para cada sub-bloque de crominancia cuadrado de tamaño 4x4. En particular, el codificador de vídeo 20 puede codificar los CBF para cada uno de los sub-bloques, y el decodificador de vídeo 30 decodifica los CBF para cada uno de los sub-bloques.

5 En otro ejemplo, si el tamaño de CU de crominancia es 8x16 y puede subdividirse adicionalmente en cuatro unidades de transformación de tamaño 4x8, para cada componente de crominancia, un indicador CBF se puede señalar en el nivel de 8x16, indicando si cualquiera de los 4 bloques de transformación de tamaño 4x8 tiene o no coeficientes distintos de cero. Si cualquiera de los 4 bloques de transformación de tamaño 4x8 tiene coeficientes distintos de cero, a continuación, para cada bloque de tamaño 4x8, puede enviarse otro indicador CBF. Un ejemplo  
10 conserva esta estructura, cambiando solamente la forma en que el CBF se señala a nivel de la hoja de RQT. En el primer ejemplo, cuando el bloque rectangular en la hoja del árbol cuádruple se divide en dos bloques de transformación cuadrados, un indicador CBF puede ser enviado para cada bloque de transformación cuadrado. Por lo tanto, en el ejemplo anterior, se envía un indicador CBF para el bloque de tamaño 8x16 y a continuación, para cada bloque de hoja de tamaño 4x8, se envían dos indicadores CBF. En otro ejemplo, para el bloque de hojas, el  
15 CBF se señala jerárquicamente. En ese caso, para cada bloque de hojas de tamaño 4x8, se envía un indicador CBF para señalar si hay o no presente algún coeficiente de transformación distinto de cero. Si este indicador es uno, se envían dos indicadores CBF adicionales, uno para cada bloque de transformación cuadrado. Hay que señalar que estos dos enfoques ejemplares se pueden usar incluso cuando los CBF de crominancia se señalan directamente (como para la luminancia) o, en lugar de eso, si los CBF de crominancia se señalan jerárquicamente.  
20 En ese caso, solo se envían los CBF para los bloques de hojas de RQT.

El componente de crominancia puede incluir componentes de crominancia primero y segundo, tales como, por ejemplo, componentes de crominancia Cr y Cb o componentes de crominancia U y V. La codificación de los sub-bloques primero y segundo incluye la codificación de los sub-bloques cuadrados primero y segundo de un primer  
25 bloque de crominancia rectangular y la codificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular.

En otro ejemplo, sin embargo, los CBF pueden codificarse jerárquicamente. En un ejemplo de codificación jerárquica de CBF de ese tipo, el codificador de vídeo 20 puede enviar un CBF para todo un bloque rectangular. Un bloque rectangular de este tipo puede dividirse en dos bloques cuadrados y el codificador de vídeo 20 puede enviar un  
30 indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero. El decodificador de vídeo 30 puede recibir un CBF para todo el bloque rectangular. Un bloque rectangular de este tipo puede dividirse en dos bloques cuadrados y el decodificador de vídeo 30 puede recibir un indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero.  
35

Más específicamente, el codificador de vídeo 20 del dispositivo de origen 12 puede codificar un CBF para todo un bloque rectangular. Un bloque rectangular de este tipo puede dividirse en dos bloques cuadrados y el codificador de vídeo 20 puede codificar un CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero. Los CBF pueden almacenarse en un dispositivo de almacenamiento 34 para un  
40 acceso posterior mediante el dispositivo de destino 14 u otros dispositivos, para decodificación y/o reproducción. Como alternativa, estos CBF pueden transmitirse, a través de la interfaz de salida 22, al enlace 16 y/o al dispositivo de almacenamiento 34. Los CBF pueden ser recibidos por la interfaz de entrada 28 del dispositivo de destino 14; puede recibir el CBF para todo el bloque rectangular y el CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero. Los CBF adicionales no se almacenarán, transmitirán ni recibirán cuando el CBF del bloque rectangular sea cero.  
45

Otro ejemplo relacionado con la codificación de datos de vídeo puede incluir la transmisión y/o recepción de un bloque de crominancia rectangular. Por ejemplo, la interfaz de salida 22 del dispositivo de origen 12 puede transmitir un bloque de crominancia rectangular, y una interfaz de entrada 28 del dispositivo de destino 14 puede recibir un  
50 bloque de crominancia rectangular. Más específicamente, el codificador de vídeo 20 del dispositivo de origen 12 puede codificar el bloque de crominancia rectangular, y el mismo puede transmitirse, a través de la interfaz de salida 22, al enlace 16 y/o almacenarse en el dispositivo de almacenamiento 34 para su posterior recuperación por un decodificador tal como el decodificador de vídeo 30. El bloque de crominancia rectangular codificado puede recibirse desde el enlace 16 y/o el dispositivo de almacenamiento 34 en la interfaz de entrada 28 del dispositivo de destino 14 y decodificarse mediante el decodificador de vídeo 30. Los datos de vídeo codificados también (o como alternativa)  
55 pueden almacenarse en el dispositivo de almacenamiento 34 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para decodificación y/o reproducción.

En algunos ejemplos de esta divulgación, el decodificador de vídeo 30 puede obtener un bloque de crominancia rectangular que se divide en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado, como se ha descrito anteriormente. La división de un bloque de crominancia rectangular en sub-bloques cuadrados, tal como se ha descrito anteriormente, puede presentar algunas consideraciones para la intra-codificación de dichos sub-bloques. Las técnicas también, o alternativamente, pueden incluir, en algunos ejemplos, muestras de intra-codificación correspondientes a un sub-bloque cuadrado de un bloque de crominancia rectangular basado en  
60 muestras reconstruidas obtenidas de la intra-codificación del otro sub-bloque cuadrado. El primer bloque de datos de crominancia intra-predichos puede ser reconstruido utilizando un primer bloque de muestras de intra-predicción y los  
65

5 primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. Por ejemplo, después de la intra-decodificación del bloque superior, el decodificador de vídeo 30 intra-decodifica el sub-bloque inferior mediante la generación de un bloque intra-predictivo y uno residual. El segundo bloque de muestras de intra-predicción puede generarse a partir del primer bloque de datos de crominancia intra-predichos. El segundo bloque de datos de crominancia intra-predichos puede ser reconstruido utilizando el segundo bloque de muestras de intra-predicción y los segundos datos residuales del segundo sub-bloque cuadrado. Por ejemplo, el decodificador de vídeo 30 puede intra-codificar el bloque inferior utilizando muestras reconstruidas del bloque superior.

10 En el caso de codificación de RQT intra-predictiva de la HEVC, las etapas 1) intra-predicción y 2) transformación - cuantización - descuantización - transformación inversa se intercalan. Según la divulgación actual, se ilustran dos ejemplos. En un ejemplo, para cada hoja rectangular en la estructura de RQT, un módulo de predicción del codificador de vídeo 20 o un módulo de predicción del decodificador de vídeo 30 puede llevar a cabo la intra-predicción en el bloque rectangular. A continuación, el módulo de predicción puede dividir el residuo de intra-predicción en dos bloques cuadrados y se aplica una transformación cuadrada a cada bloque. En otro ejemplo, el proceso de predicción y transformación está intercalado como en el caso de perfil principal de la HEVC, por ejemplo, como se ha descrito anteriormente. En este caso, un módulo de predicción del codificador de vídeo 20 o un módulo de predicción del decodificador de vídeo 30 puede dividir el bloque de hojas rectangular en dos bloques cuadrados (superior e inferior) u obtener un bloque de hojas rectangular que se divide en dos bloques cuadrados. Los módulos de predicción del codificador de vídeo 20 o del decodificador de vídeo 30 pueden llevar a cabo la intra-predicción en el sub-bloque cuadrado superior. Por ejemplo, el codificador de vídeo 20 puede intra-codificar el sub-bloque superior mediante la generación de un bloque intra-predicho. El residuo de intra-predicción se somete a transformación, cuantización, cuantización inversa y transformación inversa. Un sumador en el codificador de vídeo 20 puede añadir el bloque resultante de vuelta al bloque de intra-predicción para formar un sub-bloque cuadrado superior reconstruido. A continuación, las muestras reconstruidas a partir del bloque superior (así como otras muestras reconstruidas) se utilizan para llevar a cabo la intra-predicción para el sub-bloque cuadrado inferior. Por ejemplo, el codificador de vídeo 20 puede intra-codificar el sub-bloque cuadrado inferior utilizando muestras reconstruidas a partir del sub-bloque cuadrado superior. En otro ejemplo, el decodificador de vídeo 30 puede realizar las funciones correspondientes después de decodificar el sub-bloque cuadrado superior. Por ejemplo, el decodificador de vídeo 30 puede reconstruir el sub-bloque cuadrado superior intra-predicho y utilizar muestras de referencia reconstruidas del sub-bloque cuadrado superior para generar datos de crominancia intra-predichos para el sub-bloque cuadrado inferior. A continuación, el decodificador de vídeo 30 puede reconstruir el sub-bloque cuadrado inferior añadiendo los datos residuales para el sub-bloque cuadrado inferior a los datos de crominancia intra-predichos para el sub-bloque cuadrado inferior.

35 La FIG. 2 es un diagrama de bloques que ilustra un codificador ejemplar de vídeo 20 que puede implementar las técnicas descritas en esta divulgación. El codificador de vídeo 20 puede realizar la intra-codificación y la inter-codificación de bloques de vídeo dentro de fragmentos de vídeo. La intra-codificación se apoya en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dada. La inter-codificación se apoya en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. La intra-modalidad (modalidad I) puede referirse a cualquiera de varias modalidades de compresión de base espacial. Las inter-modalidades, tales como la predicción unidireccional (modalidad P) o la bi-predicción (modalidad B), pueden referirse a cualquiera de varias modalidades de compresión de base temporal.

45 En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye un módulo de predicción 41, una memoria de imágenes de referencia 64, un sumador 50, un módulo de transformación 52, un módulo de cuantización 54 y un módulo de codificación por entropía 56. El módulo de predicción 41 incluye un módulo de estimación de movimiento 42, un módulo de compensación de movimiento 44 y un módulo de intra-predicción 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además un módulo de cuantización inversa 58, una unidad de transformación inversa 60 y un sumador 62. También puede incluirse un filtro de desbloqueo (no mostrado en la FIG. 2) para filtrar límites de bloque, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará habitualmente la salida del sumador 62. También pueden usarse filtros de bucle adicionales (en bucle o tras el bucle), además del filtro de desbloqueo.

55 Como se muestra en la FIG. 2, el codificador de vídeo 20 recibe datos de vídeo y el módulo de predicción 41 divide los datos en bloques de vídeo. En algunos casos, el módulo de predicción 41 puede dividir los datos de vídeo en base a un análisis de la distorsión de la frecuencia. Los datos de vídeo recibidos pueden formatearse según cualquiera de los formatos de muestra descritos anteriormente. Por ejemplo, los datos de vídeo pueden formatearse según el formato de muestra 4:2:2. La división puede incluir división de datos de vídeo en fragmentos, sectores u otras unidades mayores, así como la división de bloques de vídeo, por ejemplo, según una estructura de árbol cuádruple de las LCU y las CU.

65 El codificador de vídeo 20 ilustra en general los componentes que codifican bloques de vídeo dentro de un fragmento de vídeo a codificar. El fragmento puede dividirse en múltiples bloques de vídeo (y, posiblemente, en conjuntos de bloques de vídeo mencionados como sectores). El módulo de predicción 41 puede seleccionar una entre una pluralidad de posibles modalidades de codificación, tal como una entre una pluralidad de modalidades de

intra-codificación, o una entre una pluralidad de modalidades de inter-codificación, para el bloque de vídeo actual, en base a resultados de errores (por ejemplo, la frecuencia de codificación y el nivel de distorsión). El módulo de predicción 41 puede proporcionar el bloque intra-codificado o inter-codificado resultante al sumador 50 para generar datos de bloques residuales, y al sumador 62 para reconstruir el bloque codificado para su uso como una imagen de referencia.

El módulo de intra-predicción 46, dentro del módulo de predicción 41, puede realizar la codificación intra-predictiva del bloque de vídeo actual con respecto a uno o más bloques contiguos en la misma trama o fragmento que el bloque actual que va a codificarse, para proporcionar compresión espacial. El módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44, dentro del módulo de predicción 41, realizan la codificación inter-predictiva del bloque de vídeo actual con respecto a uno o más bloques predictivos en una o más imágenes de referencia, para proporcionar compresión temporal.

El módulo de estimación de movimiento 42 puede configurarse para determinar la modalidad de inter-predicción para un fragmento de vídeo, según un patrón predeterminado para una secuencia de vídeo. El patrón predeterminado puede designar fragmentos de vídeo en la secuencia como fragmentos P o fragmentos B. El módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44 pueden estar sumamente integrados, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada mediante el módulo de estimación de movimiento 42, es el proceso de generar vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual, con respecto a un bloque predictivo dentro de una imagen de referencia.

Un bloque predictivo, para la inter-codificación, puede ser un bloque del que se descubre que se corresponde estrechamente con la PU del bloque de vídeo a codificar, 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. Como alternativa, un bloque predictivo, para la intra-codificación, puede ser un bloque que se forma sobre la base de la predicción espacial con respecto a los valores de píxel a partir de uno o más bloques contiguos. En algunos ejemplos, para la inter-predicción, el codificador de vídeo 20 puede calcular valores para posiciones de fracciones de píxel de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, el módulo de estimación de movimiento 42 puede realizar una búsqueda de movimiento con respecto a las posiciones de píxeles completos y a las posiciones de fracciones de píxel, y emitir un vector de movimiento con una precisión de fracciones de píxel.

El módulo 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 imágenes de referencia 64. El módulo de estimación de movimiento 42 envía el vector de movimiento calculado al módulo de codificación por entropía 56 y al módulo de compensación de movimiento 44.

La compensación de movimiento, realizada por el módulo de compensación de movimiento 44, puede implicar extraer o generar el bloque predictivo en base al vector de movimiento determinado por la estimación de movimiento, realizando posiblemente interpolaciones hasta la precisión de sub-píxeles. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, el módulo de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia.

El codificador de vídeo 20 forma un bloque de vídeo residual, para inter-codificación o intra-codificación, restando los valores de píxeles del bloque predictivo a los valores de píxeles del bloque de vídeo actual que está codificándose, generando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencias, tanto de luminancia como de crominancia. El sumador 50 representa el componente o los componentes que realizan esta operación de sustracción. El módulo de compensación de movimiento 44 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo, para su uso por parte del decodificador de vídeo 30 en la decodificación de los bloques de vídeo del fragmento de vídeo.

El módulo de intra-predicción 46 puede intra-predicir un bloque actual, como alternativa a la inter-predicción llevada a cabo por el módulo de estimación de movimiento 42 y el módulo de compensación de movimiento 44, como se ha descrito anteriormente. En particular, el módulo de intra-predicción 46 puede determinar una modalidad de intra-predicción a usar para codificar un bloque actual. En algunos ejemplos, el módulo de intra-predicción 46 puede codificar un bloque actual usando varias modalidades de intra-predicción, por ejemplo, durante diferentes pasadas de codificación, y el módulo de intra-predicción 46 (o el módulo de selección de modalidad 40, en algunos ejemplos) puede seleccionar una modalidad adecuada de intra-predicción a usar, entre las modalidades probadas.

Por ejemplo, el módulo de intra-predicción 46 puede calcular valores de velocidad y distorsión usando un análisis de distorsión de velocidad para las diversas modalidades de intra-predicción probadas, y seleccionar la modalidad de intra-predicción que tenga las mejores características de velocidad y distorsión entre las modalidades probadas. El análisis de velocidad y distorsión determina generalmente una magnitud 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 de bits (es decir, el número de bits) usada para generar el bloque codificado. El módulo de intra-predicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modalidad de intra-predicción presenta el mejor valor de velocidad y distorsión para el bloque. Debe tenerse en cuenta que el análisis de velocidad y distorsión se puede realizar en una combinación de los componentes de color.

En cualquier caso, tras seleccionar una modalidad de intra-predicción para un bloque, el módulo de intra-predicción 46 puede proporcionar información que indica la modalidad de intra-predicción seleccionada para el bloque, al módulo de codificación por entropía 56. El módulo de codificación por entropía 56 puede codificar la información que indica la modalidad de intra-predicción seleccionada, según las técnicas de esta divulgación. 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 modalidad de intra-predicción y una pluralidad de tablas de índices de modalidad de intra-predicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones de la modalidad de intra-predicción más probable, una tabla de índices de modalidad de intra-predicción y una tabla modificada de índices de modalidad de intra-predicción, a usar en cada uno de los contextos. El flujo de bits también (o como alternativa) puede almacenarse en el dispositivo de almacenamiento 34 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para decodificación y/o reproducción.

Después de que el módulo de predicción 41 genera el bloque predictivo para el bloque de vídeo actual, ya sea mediante la inter-predicción o la intra-predicción, el codificador de vídeo 20 forma un bloque de vídeo residual restando el bloque predictivo al bloque de vídeo actual. Los datos de vídeo residuales en el bloque residual pueden ser incluidos en una o más TU y aplicados al módulo de transformación 52. Téngase en cuenta que el módulo de transformación 52 se refiere a un componente, módulo o unidad funcional del codificador de vídeo 20, y no debería confundirse con una TU, que es una unidad de datos básica para el proceso de transformación y cuantización. El módulo de transformación 52 transforma los datos de vídeo residuales en coeficientes de transformación residuales usando una transformación, tal como una transformación de coseno discreta (DCT) o una transformación conceptualmente similar. El módulo de transformación 52 puede convertir los datos de vídeo residuales, desde un dominio de píxeles a un dominio de transformación, tal como un dominio de frecuencia. El módulo de transformación 52 puede enviar los coeficientes de transformación resultantes al módulo de cuantización 54.

En algunos ejemplos, las CU que están inter-codificadas o intra-codificadas producen componentes de luminancia cuadrados y componentes de crominancia rectangulares cuando se utiliza el formato 4:2:2. El módulo de transformación 52 puede generar sub-bloques cuadrados para las TU de crominancia rectangulares. La división de un bloque de crominancia rectangular en sub-bloques cuadrados, tal como se ha descrito anteriormente, puede presentar cierta consideración para la intra-codificación de dichos sub-bloques. Las técnicas también, o alternativamente, pueden incluir, en algunos ejemplos, muestras de intra-codificación correspondientes a un sub-bloque cuadrado de un bloque de crominancia rectangular basado en muestras reconstruidas obtenidas de la intra-codificación del otro sub-bloque cuadrado.

Como se ha descrito anteriormente, el codificador de vídeo 20 puede codificar dos CBF, un CBF para cada uno de dos sub-bloques cuadrados de un bloque de crominancia rectangular de coeficientes de transformación, directamente, sin un indicador CBF para el bloque rectangular. Por ejemplo, el módulo de codificación por entropía 56 dentro del codificador de vídeo 20 puede codificar dos CBF, un CBF para cada uno de dos sub-bloques para un bloque componente de crominancia rectangular, directamente, sin codificar un indicador CBF para un bloque rectangular. En algunos ejemplos, el módulo de transformación 52 puede dividir un bloque de crominancia rectangular de coeficientes de transformación, por ejemplo, en una hoja de árbol cuádruple, en sub-bloques cuadrados primero y segundo. En los ejemplos que incluyen un módulo de RQT o un bloque de RQT, tal módulo o bloque puede llevar a cabo la división de un bloque de crominancia rectangular de coeficientes de transformación en sub-bloques cuadrados primero y segundo. El módulo de codificación por entropía 56, u otra unidad funcional del codificador de vídeo 20, pueden codificar un CBF como un elemento sintáctico para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de codificación por entropía 56 también puede codificar un CBF como un elemento sintáctico para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de codificación por entropía 56 generalmente no codifica un CBF para el bloque de crominancia rectangular, es decir, la combinación de los sub-bloques cuadrados primero y segundo. Los CBF pueden proporcionarse en el flujo de bits codificado para indicar si un sub-bloque dado incluye o no incluye coeficientes de transformación distintos de cero. Si el CBF indica que un sub-bloque incluye coeficientes de transformación distintos de cero, un decodificador de vídeo 30 analiza sintácticamente el sub-bloque para procesar los coeficientes de transformación distintos de cero. De lo contrario, si el CBF no indica que el sub-bloque incluye

coeficientes de transformación distintos de cero, el decodificador de vídeo 30 puede omitir el análisis sintáctico del sub-bloque para el procesamiento de coeficientes de transformación.

5 En un ejemplo específico, si el tamaño de la CU de crominancia es 8x16 y se subdivide en 4 unidades de transformación de tamaño 4x8, para cada componente de crominancia, el módulo de codificación por entropía 56 por lo general no codificará un CBF, y por lo general no se señalará un CBF en el nivel de bloque de tamaño 4x8. El bloque de tamaño 4x8 puede además subdividirse en dos sub-bloques de tamaño 4x4. El módulo de codificación por entropía 56 puede codificar un CBF para cada bloque de tamaño 4x4. En un ejemplo, el componente de crominancia puede incluir componentes de crominancia primero y segundo, por ejemplo, Cb y Cr o U y V. La codificación de los sub-bloques primero y segundo incluye la codificación de los sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la codificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular.

15 En otro ejemplo, sin embargo, el módulo de codificación por entropía 56 puede codificar los CBF jerárquicamente. En dicho ejemplo de codificación jerárquica de los CBF, el codificador de vídeo 20 del dispositivo de origen 12 puede enviar un CBF para todo un bloque rectangular, por ejemplo, el módulo de codificación por entropía 56 puede codificar un CBF para todo un bloque rectangular. Tal bloque rectangular se puede dividir en dos bloques cuadrados y el dispositivo de origen 12 puede enviar un indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero, por ejemplo, el codificador de vídeo 20 puede codificar un indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero.

25 Otro ejemplo relacionado con la codificación de datos de vídeo puede incluir la transmisión de un bloque de crominancia rectangular. Por ejemplo, el dispositivo de origen 12 puede transmitir un bloque de crominancia rectangular. Más específicamente, el módulo de codificación por entropía 56 del codificador de vídeo 20 puede codificar el bloque de crominancia rectangular, y el mismo puede transmitirse a través de la interfaz de salida 22 hasta el enlace 16. Como alternativa, el codificador de vídeo 20 puede codificar el bloque de crominancia rectangular y el mismo puede almacenarse en el dispositivo de almacenamiento 34, por ejemplo, en algún tipo de medios o memoria de almacenamiento. En general, muchos de tales bloques de crominancia rectangulares se almacenarían utilizando el dispositivo de almacenamiento 34, por ejemplo, para formar una imagen completa o una serie completa de imágenes. El bloque o bloques de crominancia rectangulares almacenados pueden posteriormente ser leídos desde un dispositivo de almacenamiento 34, por ejemplo, independientemente de medios cualesquiera en los cuales se haya almacenado el bloque de crominancia rectangular. Se entenderá que, en algunos ejemplos, un dispositivo de almacenamiento diferente 34 puede ser utilizado por el dispositivo de origen 12 para almacenar el bloque de crominancia rectangular procedente del dispositivo de almacenamiento utilizado para recuperar el bloque de crominancia rectangular mediante el dispositivo de destino 14.

40 En algunos ejemplos, el decodificador de vídeo 30 puede dividir un bloque de crominancia rectangular en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. El decodificador de vídeo 30 intra-decodifica el sub-bloque superior mediante la generación de un bloque intra-predictivo y uno residual. Como se ha descrito anteriormente, el primer sub-bloque cuadrado de los datos de crominancia intra-predichos puede reconstruirse utilizando un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. Por ejemplo, el decodificador de vídeo 30 puede intra-decodificar un sub-bloque cuadrado inferior de un bloque de crominancia rectangular utilizando muestras reconstruidas del sub-bloque cuadrado superior como muestras de intra-predicción. Los datos de crominancia intra-predichos para el segundo sub-bloque cuadrado se pueden generar a partir de muestras de referencia obtenidas a partir del primer sub-bloque cuadrado de datos de crominancia. El segundo sub-bloque cuadrado de datos de crominancia se puede reconstruir usando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado.

50 En algunos ejemplos, la codificación de los sub-bloques primero y segundo en el codificador de vídeo 20 puede incluir la codificación de los sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la decodificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular. En algunos ejemplos, el primer sub-bloque puede ser un sub-bloque superior y el segundo sub-bloque puede ser un sub-bloque inferior del bloque de crominancia rectangular, como se ilustra en la FIG. 8A. En algunos ejemplos, el bloque de crominancia rectangular tiene un formato de muestreo 4:2:2, como se ilustra en las FIGs. 4B y 5.

60 En algunos ejemplos, una parte del codificador de vídeo 20, por ejemplo, el módulo de cuantización inversa 58 y/o el módulo de transformación inversa 60 pueden decodificar los datos de vídeo. Por ejemplo, el codificador de vídeo 20 puede, por ejemplo, en el módulo de cuantización inversa 58 y/o el módulo de transformación inversa 60, recibir un bloque de crominancia rectangular, en donde el bloque de crominancia rectangular se divide en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. El módulo de cuantización inversa 58 y/o el módulo de transformación inversa 60 pueden reconstruir un primer bloque de datos de crominancia usando un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. El primer bloque reconstruido de datos de crominancia se puede almacenar en la memoria de imágenes de referencia 64. El módulo de predicción 41 puede generar un segundo bloque de datos de crominancia intra-

predichos para el segundo sub-bloque, en base a muestras de referencia obtenidas a partir del primer bloque reconstruido de datos de crominancia, y generar datos residuales basados en las diferencias entre el segundo sub-bloque y los datos de crominancia intra-predichos. Por lo tanto, el segundo sub-bloque se intra-predice usando muestras de referencia de intra-predicción a partir de la reconstrucción del primer sub-bloque. El módulo de cuantización inversa 58 y/o el módulo de transformación inversa 60 pueden reconstruir el segundo sub-bloque de datos de crominancia utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado, por ejemplo, para su almacenamiento en la memoria de imágenes de referencia 64. En otros ejemplos, otras partes del codificador de vídeo 20 pueden realizar algunas de, o todas, estas funciones.

En un ejemplo, el codificador de vídeo 20, por ejemplo, el módulo de cuantización inversa 58 y/o módulo de transformación inversa 60, cuantifica inversamente los coeficientes de transformación del primer sub-bloque cuadrado, transforma inversamente los coeficientes de transformación cuantizados de forma inversa, para generar los primeros datos residuales, y añade los primeros datos residuales al primer bloque de datos de crominancia intra-predichos para reconstruir el primer bloque de datos de crominancia. El codificador de vídeo 20, por ejemplo, en el módulo de predicción 41, a continuación, intra-codifica el segundo sub-bloque cuadrado de datos de crominancia utilizando muestras de referencia de intra-predicción a partir del primer sub-bloque reconstruido de datos de crominancia.

En algunos ejemplos, el codificador de vídeo 20 puede intra-codificar el segundo sub-bloque cuadrado de datos de crominancia utilizando muestras de referencia de intra-predicción, a partir de un bloque contiguo espacial adicional, por ejemplo, tal como un bloque contiguo izquierdo, ya sea solo o en combinación con muestras de referencia de intra-predicción a partir del primer sub-bloque cuadrado reconstruido. Como se ha mencionado anteriormente, en algunos ejemplos, el bloque contiguo adicional puede ser un bloque contiguo izquierdo. En otros ejemplos, se pueden usar otros bloques contiguos.

La FIG. 3 es un diagrama de bloques que ilustra un decodificador ejemplar de vídeo 30 que puede implementar las técnicas descritas en esta divulgación. En el ejemplo de la FIG. 3, el decodificador de vídeo 30 incluye un módulo de decodificación por entropía 80, un módulo de predicción 81, un módulo de cuantización inversa 86, un módulo de transformación inversa 88, un sumador 90 y una memoria de imágenes de referencia 92. El módulo de predicción 81 incluye un módulo de compensación de movimiento 82 y un módulo de intra-predicción 84. En algunos ejemplos, el decodificador de vídeo 30 puede realizar una pasada de decodificación generalmente recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 de la FIG. 2.

Durante el proceso de decodificación, el decodificador de vídeo 30 recibe un flujo de bits de vídeo codificado que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. El módulo de decodificación por entropía 80 del decodificador de vídeo 30 decodifica por entropía el flujo de bits para generar coeficientes cuantizados, vectores de movimiento y otros elementos sintácticos. El módulo de decodificación por entropía 80 remite los vectores de movimiento y otros elementos sintácticos al módulo de predicción 81. El decodificador de vídeo 30 puede recibir los elementos sintácticos al nivel del fragmento de vídeo y/o al nivel del bloque de vídeo.

Cuando el fragmento de vídeo se codifica como un fragmento intra-codificado (I), el módulo de intra-predicción 84 del módulo de predicción 81 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual, en base a una modalidad de intra-predicción señalizada, y datos de bloques previamente decodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento inter-codificado (es decir, B o P), el módulo de compensación de movimiento 82 del módulo de predicción 81 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual, en base a los vectores de movimiento y a otros elementos sintácticos recibidos desde el módulo de decodificación por entropía 80. Los bloques predictivos pueden ser generados a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El decodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por omisión, en base a las imágenes de referencia almacenadas en la memoria de imágenes de referencia 92.

El módulo de compensación de movimiento 82 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual, analizando sintácticamente los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos para el bloque de vídeo actual que está siendo decodificado. Por ejemplo, el módulo de compensación de movimiento 82 usa algunos de los elementos sintácticos recibidos para determinar una modalidad de predicción (por ejemplo, una entre una pluralidad de modalidades de inter-predicción), usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de inter-predicción (por ejemplo, fragmento B o fragmento P), información de construcción para una o más de las listas de imágenes de referencia para el fragmento, vectores de movimiento para cada bloque de vídeo inter-codificado del fragmento, estado de inter-predicción para cada bloque de vídeo inter-codificado del fragmento y otra información para decodificar los bloques de vídeo en el fragmento de vídeo actual.

El módulo de compensación de movimiento 82 también puede realizar la interpolación en base a filtros de interpolación. El módulo de compensación de movimiento 82 puede usar filtros de interpolación como los usados por

el codificador de vídeo 20 durante la codificación de los bloques de vídeo para calcular valores interpolados para píxeles sub-enteros de bloques de referencia. En este caso, el módulo de compensación de movimiento 82 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques predictivos.

El módulo de cuantización inversa 86 cuantiza de manera inversa, es decir, descuantiza, los coeficientes de transformación cuantizados, proporcionados en el flujo de bits y decodificados por el módulo de decodificación por entropía 80. El proceso de cuantización inversa puede incluir el uso de un parámetro de cuantización calculado por el codificador de vídeo 20 para cada bloque de vídeo en el fragmento de vídeo, para determinar un grado de cuantización y, asimismo, un grado de cuantización inversa que debería aplicarse.

El módulo de transformación inversa 88 recibe coeficientes de transformación inversa y aplica una transformación inversa, por ejemplo, una DCT inversa, una transformación inversa entera o un proceso de transformación inversa conceptualmente similar, a los coeficientes de transformación, con el fin de generar bloques residuales en el dominio de píxeles. En algunos ejemplos, el módulo de transformación inversa 88 puede recibir coeficientes de transformación que fueron generados por un codificador de vídeo en base a las técnicas de división de la unidad de transformación.

Después de que el módulo de compensación de movimiento 82 o el módulo de intra-predicción 84 genere el bloque predictivo para el bloque de vídeo actual, en base a los vectores de movimiento y a otros elementos sintácticos, el decodificador de vídeo 30 forma un bloque de vídeo decodificado sumando los bloques residuales procedentes del módulo de transformación inversa 88 a los correspondientes bloques predictivos generados por el módulo de compensación de movimiento 82. El sumador 90 representa el componente o los componentes que llevan a cabo esta operación de suma.

Si se desea, también puede aplicarse un filtro de desbloqueo para filtrar los bloques decodificados, con el fin de eliminar distorsiones de efecto pixelado. Otros filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) también pueden ser usados para allanar las transiciones de píxeles, o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo decodificados en una trama o imagen dada son a continuación almacenados en la memoria de imágenes de referencia 92, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La memoria de imágenes de referencia 92 almacena también vídeo decodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1.

Como se ha descrito anteriormente, el decodificador de vídeo 30 puede decodificar dos CBF, incluyendo un primer CBF para un sub-bloque cuadrado y un segundo CBF para un segundo sub-bloque cuadrado de un bloque de crominancia rectangular de coeficientes de transformación. Por lo tanto, el decodificador de vídeo 30 puede recibir los CBF para los sub-bloques cuadrados directamente, sin recibir un indicador CBF para un bloque rectangular en el que residen los sub-bloques. Por ejemplo, el módulo de decodificación por entropía 80 dentro del decodificador de vídeo 30 puede decodificar dos CBF, un CBF para cada uno de dos bloques, directamente, sin un indicador CBF para un bloque rectangular. Como se ha descrito anteriormente, en algunos ejemplos, un módulo en el codificador de vídeo 20, tal como el módulo de predicción 41, puede dividir un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, en sub-bloques cuadrados primero y segundo. En consecuencia, el decodificador de vídeo 30 puede recibir los CBF y coeficientes correspondientes a dos sub-bloques cuadrados. El módulo de decodificación por entropía 80 puede decodificar un CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de decodificación por entropía 80 también puede decodificar un CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de decodificación por entropía 80 generalmente no decodifica un CBF para el bloque de crominancia rectangular, es decir, la combinación de los sub-bloques cuadrados primero y segundo, ya que no se recibe un CBF para el bloque rectangular en este ejemplo.

En un ejemplo específico, si el tamaño de la CU de crominancia es 8x16 y se subdivide en 4 unidades de transformación de tamaño 4x8, para cada componente de crominancia, el módulo de decodificación por entropía 80 generalmente no decodificará un CBF, y en general no se señalará un CBF, en el nivel del bloque rectangular de tamaño 4x8. El bloque de tamaño 4x8 puede subdividirse adicionalmente en dos sub-bloques cuadrados de tamaño 4x4. El módulo de decodificación por entropía 80 puede decodificar un CBF para cada bloque de tamaño 4x4. En un ejemplo, el componente de crominancia incluye componentes de crominancia primero y segundo. La decodificación de los sub-bloques primero y segundo incluye la decodificación de sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la decodificación de sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular. Normalmente, un CBF se utiliza para señalar la presencia o ausencia de coeficientes distintos de cero. En algunos ejemplos de esta divulgación, el codificador de vídeo 20 señala dos CBF para sub-bloques de crominancia, un CBF para cada uno de dos sub-bloques de crominancia cuadrados de un bloque rectangular de coeficientes de transformación, directamente, sin enviar un indicador CBF para el bloque rectangular. A continuación, el decodificador de vídeo 30 puede recibir dos CBF, un CBF para cada uno de dos sub-bloques de crominancia cuadrados. No se señala ningún CBF en el flujo de bits codificado por el codificador de vídeo 20 (en este ejemplo) para el bloque rectangular, y por tanto no se recibirá ningún CBF para el bloque de

5 crominancia rectangular en el flujo de bits codificado por el decodificador de vídeo 30. Si el CBF indica que un sub-bloque incluye coeficientes de transformación distintos de cero, un decodificador de vídeo 30 analiza sintácticamente el sub-bloque para procesar los coeficientes de transformación distintos de cero. De lo contrario, si el CBF no indica que el sub-bloque incluye coeficientes de transformación distintos de cero, el decodificador de vídeo 30 puede omitir el análisis sintáctico del sub-bloque para el procesamiento de coeficientes de transformación.

10 En otro ejemplo, sin embargo, el módulo de decodificación por entropía 80 puede decodificar los CBF jerárquicamente. En un ejemplo de ese tipo de decodificación jerárquica de los CBF, el decodificador de vídeo 30 del dispositivo de destino 14 puede recibir un CBF para todo un bloque rectangular, por ejemplo, el módulo de decodificación por entropía 80 puede codificar un CBF para todo un bloque rectangular. Un bloque rectangular de este tipo puede haberse dividido en dos bloques cuadrados, por ejemplo, en un codificador de vídeo 20, y el codificador de vídeo 20 del dispositivo de origen 12 puede haber enviado un indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular sea distinto a cero; por ejemplo, el módulo de decodificación por entropía 80 puede decodificar un indicador CBF adicional para cada uno de los bloques cuadrados cuando el CBF del bloque rectangular es distinto de cero. En este caso, el CBF para el bloque rectangular indica si cualquiera de los sub-bloques incluye o no coeficientes distintos de cero. Si es así, entonces el decodificador de vídeo 30 recibe los CBF de los sub-bloques individuales, indicando si cada sub-bloque respectivo incluye o no coeficientes distintos de cero.

20 El decodificador de vídeo 30 representa un ejemplo de un decodificador configurado para recibir y procesar un bloque de crominancia rectangular que se ha dividido en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado, por ejemplo, en el codificador de vídeo 20. Como se ha descrito anteriormente, el primer sub-bloque cuadrado de los datos de crominancia intra-predichos puede reconstruirse utilizando un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. Un segundo bloque de datos de crominancia intra-predichos para el segundo sub-bloque puede generarse a partir del primer bloque de datos de crominancia intra-predichos. El segundo sub-bloque de datos de crominancia se puede reconstruir usando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado.

30 Como se ha expuesto anteriormente, en algunos ejemplos, el decodificador de vídeo 30 representa un ejemplo de un decodificador configurado para obtener un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, es decir, en sub-bloques cuadrados primero y segundo. El decodificador de vídeo 30, por ejemplo, el módulo de decodificación por entropía 80, puede decodificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de decodificación por entropía 80 en el decodificador de vídeo 30 también puede decodificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. Como se ha descrito en la presente memoria, en algunos ejemplos, el decodificador de vídeo 30 generalmente no decodifica un CBF para el bloque de crominancia rectangular.

40 En algunos ejemplos, el componente de crominancia puede incluir componentes de crominancia primero y segundo. La decodificación de los sub-bloques cuadrados primero y segundo en el decodificador de vídeo 30 puede incluir la decodificación de los sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la decodificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular. 45 En algunos ejemplos, el primer sub-bloque puede ser un sub-bloque cuadrado superior y el segundo sub-bloque puede ser un sub-bloque cuadrado inferior del bloque de crominancia rectangular, como se ilustra en la FIG. 8A. En algunos ejemplos, el bloque de crominancia rectangular tiene un formato de muestreo 4:2:2, como se ilustra en las FIGs. 4B y 5.

50 En algunos ejemplos, una parte del decodificador de vídeo 30, por ejemplo, el módulo de decodificación por entropía 80, puede decodificar por entropía datos de vídeo. Por ejemplo, el módulo de decodificación por entropía 80 puede recibir un bloque de crominancia rectangular, en donde el bloque de crominancia rectangular se divide en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. El módulo de predicción 81 puede reconstruir un primer sub-bloque cuadrado de datos de crominancia usando un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. El módulo de predicción 81 puede generar un segundo bloque de datos de crominancia intra-predichos para el segundo sub-bloque cuadrado, en base a muestras de referencia en el primer sub-bloque cuadrado reconstruido, y reconstruir un segundo sub-bloque cuadrado de datos de crominancia intra-predichos utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado. En otros ejemplos, otras partes del decodificador de vídeo 30 pueden realizar algunas de, o todas, estas funciones. Por lo tanto, las muestras de referencia intra-predictivas del sub-bloque cuadrado superior se pueden utilizar para generar el bloque intra-predicho para el sub-bloque cuadrado inferior de un bloque de crominancia rectangular.

65 En un ejemplo, el decodificador de vídeo 30, por ejemplo, el módulo de cuantización inversa 86 y/o el módulo de transformación inversa 88, cuantifica de forma inversa los coeficientes de transformación del primer sub-bloque cuadrado, y transforma de forma inversa los coeficientes de transformación cuantizados de forma inversa, para

generar los primeros datos residuales. El sumador 90 puede agregar los primeros datos residuales al primer bloque de datos de crominancia intra-predichos, para reconstruir el primer sub-bloque cuadrado de datos de crominancia.

En algunos ejemplos, el decodificador de vídeo 30 puede reconstruir el segundo sub-bloque de datos de crominancia mediante la generación de un bloque de datos de crominancia intra-predichos, utilizando muestras de referencia de intra-predicción de un bloque espacialmente contiguo adicional, ya sea solo o en combinación con muestras de referencia de intra-predicción del primer sub-bloque cuadrado, por ejemplo, el sub-bloque cuadrado superior. En algunos ejemplos, el bloque contiguo adicional puede ser un bloque contiguo izquierdo. En otros ejemplos, se pueden usar otros bloques contiguos.

Las FIGs. 4A a 4C son diagramas conceptuales que ilustran diferentes formatos de muestra para los componentes de luminancia y de crominancia de una unidad de codificación. La FIG. 4A es un diagrama conceptual que ilustra el formato de muestra 4:2:0. Como se ilustra en la FIG. 4A, para el formato de muestra 4:2:0, los componentes de crominancia tienen un cuarto del tamaño del componente de luminancia. Así, para una CU formateada según el formato de muestra 4:2:0, hay cuatro muestras de luminancia para cada muestra de un componente de crominancia. La FIG. 4B es un diagrama conceptual que ilustra el formato de muestra 4:2:2. Como se ilustra en la FIG. 4B, para el formato de muestra 4:2:2, los componentes de crominancia tienen la mitad del tamaño del componente de luminancia. Así, para una CU formateada según el formato de muestra 4:2:2, hay dos muestras de luminancia por cada muestra de un componente de crominancia. La FIG. 4C es un diagrama conceptual que ilustra el formato de muestra 4:4:4. Como se ilustra en la FIG. 4C, para el formato de muestra 4:4:4, los componentes de crominancia tienen el mismo tamaño del componente de luminancia. Así, para una CU formateada según el formato de muestra 4:4:4, hay una muestra de luminancia para cada muestra de un componente de crominancia.

La FIG. 5 es un diagrama conceptual que ilustra un ejemplo de una unidad de codificación de tamaño 16x16, formateada según un formato de muestra 4:2:2. Como se ha descrito anteriormente, una CU se define habitualmente según el número de muestras de luminancia horizontales y verticales. A continuación, como se ilustra en la FIG. 5, una CU de tamaño 16x16, formateada según el formato de muestra 4:2:2, incluye muestras de tamaño 16x16 de componentes de luminancia y muestras de tamaño 8x16 para cada componente de crominancia. Además, como se ha descrito anteriormente, una CU puede dividirse en CU más pequeñas. Por ejemplo, la CU ilustrada en la FIG. 5 puede dividirse en cuatro CU de tamaño 8x8, donde cada CU incluye muestras de tamaño 8x8 para el componente de luminancia y muestras de tamaño 4x8 para cada componente de crominancia.

Además, en algunas normas de codificación de vídeo, una unidad de codificación puede dividirse en unidades más pequeñas con fines de predicción o transformación. Según la HEVC, una CU puede incluir una o más unidades de predicción (PU) y/o una o más unidades de transformación (TU). Esta divulgación también usa el término "bloque", "partición" o "parte" para referirse a una CU, una PU o una TU. En general, "parte" puede referirse a cualquier subconjunto de una trama de vídeo. Además, esta divulgación usa habitualmente el término "bloque de vídeo" para referirse a un nodo de codificación de una CU. En algunos casos específicos, esta divulgación también puede usar el término "bloque de vídeo" para referirse a un bloque arbolado, es decir, una LCU, o una CU, que incluye un nodo de codificación y las PU y TU. Por lo tanto, un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU y los bloques de vídeo pueden tener tamaños fijos o variables, y pueden diferir en tamaño según una norma de codificación especificada.

Una PU puede representar la totalidad, o una parte, de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Las PU pueden tener formas cuadradas o rectangulares. Por ejemplo, si la CU en la FIG. 5 se codifica utilizando una inter-predicción, se puede dividir en cuatro PU rectangulares, donde cada PU incluye información que identifica un conjunto de muestras de referencia en una trama temporalmente adyacente. Los conjuntos de muestras de referencia se pueden combinar para formar un bloque de vídeo predictivo. Como se ha descrito anteriormente, el bloque de vídeo predictivo puede restarse de la CU para formar datos residuales.

Como se ha descrito anteriormente, una transformación se puede aplicar a los datos residuales para transformar los datos residuales, desde un dominio de píxeles a un dominio de transformación. Un bloque de transformación o TU puede corresponder a un conjunto de datos residuales a los que se aplica una transformación. Las TU representan el tamaño de un conjunto de valores de diferencia de píxel, con el fin de realizar una transformación y generar un conjunto correspondiente de coeficientes de transformación. El tamaño de una TU puede ser el mismo que el tamaño de una CU, o una CU puede dividirse en una pluralidad de TU. Por ejemplo, en la CU ilustrada en la FIG. 5 puede realizarse una transformación sobre los valores residuales asociados a la matriz de tamaño 16x16 de muestras de luminancia, o se puede realizar una transformación sobre cada una de cuatro formaciones de tamaño 8x8 de muestras de luminancia. Las TU más grandes generalmente proporcionan mayor compresión, con un "efecto pixelado" más perceptible en una imagen reconstruida, mientras que las TU más pequeñas generalmente proporcionan menos compresión con un "efecto pixelado" menos perceptible. La selección de tamaños de TU se puede basar en el análisis de optimización de velocidad y distorsión.

De forma similar a una LCU, una TU puede dividirse de forma recursiva en TU más pequeñas. Las TU que resultan de la división de una TU en TU más pequeñas pueden denominarse una estructura de bloques de transformación.

Un ejemplo de una estructura de bloques de transformación es la llamada estructura de árbol. La estructura de árbol puede codificar un bloque de transformación, ya sea como una TU completa o como dividida en una serie de TU más pequeñas. Este proceso se puede hacer de forma recursiva para cada bloque en cada nivel de descomposición diferente.

5 La FIG. 6 es un diagrama conceptual que ilustra una estructura de descomposición en árbol cuádruple, donde una CU se divide, o no se divide, en cuatro bloques de un tamaño de una cuarta parte, en niveles sucesivos. En la FIG. 6, todo el bloque puede corresponder a la CU original. Las líneas discontinuas indican un resultado de la descomposición de bloques de transformación según una estructura de árbol cuádruple. Cabe señalar que la descomposición ilustrada en la FIG. 6 es uno de varios resultados posibles de la descomposición. Como se ilustra en la FIG. 6, hay tres niveles de descomposiciones de transformación. En el primer nivel (es decir, la descomposición de nivel-1), todo el bloque de transformación se divide en cuatro bloques del tamaño de una cuarta parte. A continuación, en el segundo nivel (es decir, el nivel-2), el segundo bloque del tamaño de una cuarta parte se vuelve a dividir en cuatro bloques de tamaño 1/16. A continuación, en el tercer nivel (es decir, el nivel-3), el cuarto bloque de tamaño 1/16 se vuelve a dividir en cuatro bloques de transformación todavía más pequeños. Durante el proceso de codificación, un codificador de vídeo puede determinar si un bloque de transformación debería o no dividirse todavía más, en base al análisis de optimización de velocidad y distorsión.

20 La técnica de descomposición ilustrada en la FIG. 6 se denomina una estructura de descomposición en árbol cuádruple, en cuyo caso un bloque se divide, o no se divide, en cuatro bloques del tamaño de una cuarta parte. En la FIG. 6, los bloques solo están divididos en sub-bloques de forma cuadrada. Sin embargo, en otras técnicas de descomposición, tales como las descritas en la HEVC WD10, un bloque puede también dividirse en sub-bloques con forma rectangular, con fines de transformación.

25 La FIG. 7 es un diagrama de nivel que ilustra una descomposición en árbol cuádruple. La FIG. 7 ilustra un ejemplo de un diagrama de nivel que puede utilizarse para señalar la descomposición en árbol cuádruple ilustrada en la FIG. 6. Por lo tanto, la FIG. 7 proporciona una forma alternativa de expresar una estructura de descomposición. Como se ilustra en la FIG. 7, en diferentes niveles de descomposición, un indicador de división de transformación puede señalizarse de la forma siguiente:

30 Nivel 0: 1  
 Nivel 1: 0, 1, 0, 0  
 Nivel 2: 0, 0, 0, 1  
 Nivel 3: 0, 0, 0, 0

35 En el nivel 0, que es el nivel de unidad de codificación, se señala un indicador de 1 porque la transformación está aún más dividida. En el nivel 1, solo el segundo bloque de la cuarta parte del tamaño está dividido adicionalmente, por lo que los indicadores de 0, 1, 0, 0 son enviados en el flujo de bits codificado por el codificador de vídeo. En el nivel 2, dado que otros bloques no están adicionalmente divididos, solo el segundo bloque de la cuarta parte del tamaño necesita ser señalado adicionalmente por un codificador de vídeo. En el segundo bloque de la cuarta parte del tamaño, solo el cuarto bloque ha de dividirse adicionalmente, por lo que se envían los indicadores de 0, 0, 0, 1 en el nivel 2, mediante el codificador de vídeo. En el nivel 3, ningún bloque ha de dividirse adicionalmente, por lo que se envían los indicadores de 0, 0, 0, 0. Para mayor claridad, se debería mencionar que, en la presente divulgación, un valor de nivel más pequeño significa un nivel más alto (es decir, un nivel más cercano al nivel de raíz) en la estructura de descomposición. Como se ilustra en la FIG. 6 y en la FIG. 7, el nivel 0 es el nivel de raíz o el nivel supremo.

50 Las FIGs. 8A a 8B ilustran diferentes casos de división de unidad de transformación para bloques de vídeo formateados según el formato de muestra 4:2:2. En cada una de las FIGs. 8A a 8B, los componentes de crominancia pueden dividirse en sub-bloques superiores e inferiores. En un ejemplo, primero, puede seguirse un árbol cuádruple de luminancia para dividir los componentes de crominancia en bloques rectangulares (por ejemplo, para el formato 4:2:2). A continuación, los bloques de crominancia de hoja pueden dividirse en sub-bloques cuadrados superiores e inferiores. Mediante la división de los bloques rectangulares para los componentes de crominancia en sub-bloques cuadrados superiores e inferiores, pueden generarse formas y tamaños de TU alternativos para los componentes de crominancia, en comparación con un caso en el que los componentes de crominancia no se dividen primero en sub-bloques superiores e inferiores. Por ejemplo, se podrían generar formas de TU cuadradas de tal manera que puedan utilizarse transformaciones cuadradas.

60 Los bloques de vídeo en las FIGs. 8A a 8B pueden corresponder a los bloques de vídeo o a las CU de cualquiera de los siguientes tamaños: 4x4, 8x8, 16x16, 32x32 y 64x64. En general, mediante la división de un componente de crominancia en sub-bloques superior e inferior, se generan el doble de las TU para el componente de crominancia en comparación con un caso en el que el componente de crominancia no se divide en sub-bloques superior e inferior. Además, mediante la división de un componente de crominancia en sub-bloques superior e inferior, se pueden generar TU donde una dimensión vertical u horizontal de una TU se divide entre dos, en comparación con el caso en el que el componente de crominancia no se divide en sub-bloques superior e inferior. Algunos casos específicos de división de una CU en TU, en los que los componentes de crominancia se dividen en sub-bloques

superior e inferior, se describen de forma detallada a continuación. Cabe señalar, sin embargo, que, en aras de la brevedad, no se han descrito de forma detallada todas las posibles combinaciones de partición de TU que pueden obtenerse a partir de la división de los componentes de crominancia en sub-bloques superior e inferior, pero las técnicas de división descritas con respecto a las FIGs. 8A a 8B se pueden aplicar a varios niveles de descomposición.

La FIG. 8A ilustra unidades ejemplares de transformación basadas en una estructura de división donde los componentes de crominancia se dividen en sub-bloques superiores e inferiores y no se aplica ninguna división adicional basada en el componente de luminancia. Como se ilustra en la FIG. 8A, un bloque de luminancia no se divide adicionalmente, y cada uno de los bloques de crominancia correspondientes se divide en un sub-bloque superior y un sub-bloque inferior del bloque de crominancia rectangular. En el ejemplo ilustrado en la FIG. 8A, la determinación de dividir aún más los componentes de crominancia después de que los bloques de crominancia están divididos en un sub-bloque superior y un sub-bloque inferior del bloque de crominancia rectangular puede estar basada en la división del componente de luminancia. De este modo, en el ejemplo ilustrado en la FIG. 8A, cada uno de los componentes de crominancia no se puede dividir más porque el bloque de luminancia no está dividido adicionalmente.

En un ejemplo, la CU asociada a la división de TU, ilustrada en la FIG. 8A, puede ser una CU de tamaño 8x8. En este caso, la división de los componentes de crominancia en sub-bloques superior e inferior da como resultado una TU de tamaño 8x8 para el componente de luminancia y dos TU de tamaño 4x4 para los componentes de crominancia. Este caso se puede contrastar con un caso en el que los componentes de crominancia no se dividen en sub-bloques superior e inferior, lo cual daría como resultado una TU de tamaño 8x8 para el componente de luminancia y una TU de tamaño 4x8 para los componentes de crominancia. Como se ha descrito anteriormente, una TU de tamaño 4x4 puede estar definida para la HEVC mientras que una TU de tamaño 4x8 tal vez no esté definida o disponible. Por lo tanto, la división de los componentes de crominancia en sub-bloques superior e inferior puede dar lugar a formas y tamaños de TU más útiles.

Una vez más, dado que la codificación de coeficientes de transformación de la HEVC no presta soporte al caso en el que todos los coeficientes son cero; normalmente, un CBF se utiliza para señalar la presencia o ausencia de coeficientes distintos de cero. En la HEVC, los CBF de crominancia pueden codificarse de forma jerárquica. En un ejemplo de ese tipo de codificación jerárquica de los CBF, puede enviarse un indicador CBF para todo el bloque rectangular. Un bloque de este tipo puede dividirse en dos bloques cuadrados y puede enviarse un indicador CBF adicional para cada uno de los bloques cuadrados, por ejemplo, solo cuando el CBF del bloque rectangular es distinto de cero.

En un ejemplo independiente, si el tamaño de CU de crominancia es 8x16 y se subdivide en cuatro unidades de transformación de tamaño 4x8, para cada componente de crominancia, se señala un CBF a nivel del bloque de tamaño 4x8. El bloque de tamaño 4x8 puede subdividirse además en dos bloques de tamaño 4x4. El CBF señalado a nivel del bloque de tamaño 4x8 indica si alguno de los bloques de transformación de tamaño 4x4 tiene o no coeficientes distintos de cero. Si alguno de los bloques de transformación de tamaño 4x4 tiene coeficientes distintos de cero, para cada bloque de tamaño 4x4 se envía otro CBF.

En general, la CU asociada a las particiones de TU ilustradas en la FIG. 8A puede describirse como una CU de tamaño N por N. La división de los componentes de crominancia en sub-bloques superior e inferior da como resultado dos TU de tamaño N/2 por N/2, a diferencia de una TU de tamaño N/2 por N en el caso en que los componentes de crominancia no se dividen en sub-bloques superior e inferior, lo cual puede permitir el uso de transformaciones cuadradas para procesar los bloques. Por lo tanto, la división de los componentes de crominancia en sub-bloques superior e inferior da como resultado dos TU cuadradas, en comparación con una TU rectangular con una razón de aspecto de 1 a 2, en el caso en que los componentes de crominancia no se dividen en sub-bloques superior e inferior. Como se ha señalado anteriormente, en este ejemplo, cada uno de los sub-bloques de crominancia tiene la misma razón de aspecto que el bloque de luminancia. Cabe señalar que, en otros ejemplos, las técnicas descritas con respecto a la FIG. 8A se pueden aplicar a las CU de tamaños 4x4, 6x16, 32x32 o 64x64. En aras de la brevedad, los correspondientes tamaños de TU de luminancia y crominancia para posibles tamaños de CU no se describen de forma detallada. Como se ha descrito en el presente documento, un codificador puede codificar los CBF para el sub-bloque superior y el sub-bloque inferior. Además, como se ha descrito en el presente documento, en otro ejemplo, un codificador puede utilizar un sub-bloque superior reconstruido para generar muestras intra-predictivas para un sub-bloque inferior del bloque de crominancia rectangular.

La FIG. 8B ilustra unidades ejemplares de transformación basadas en una estructura de división donde se aplica la división en árbol cuádruple y los componentes de crominancia se dividen en sub-bloques superior e inferior. Como se ilustra en la FIG. 8B, un bloque de luminancia se divide en cuatro sub-bloques de forma cuadrada con fines de transformación. Cada uno de los bloques de crominancia correspondientes se divide en un sub-bloque superior y un sub-bloque inferior, que son ambos cuadrados, y a continuación cada uno de los sub-bloques superior e inferior se divide además en cuatro bloques de forma cuadrada con tamaños todavía más pequeños. En general, la CU en la FIG. 8B puede describirse como una CU de tamaño N por N. La división del componente de luminancia da como resultado cuatro TU de tamaño N/2 por N/2. La división de los componentes de crominancia en sub-bloques superior

e inferior da como resultado ocho TU de tamaño N/4 por N/4, en lugar de cuatro TU de tamaño N/4 por N/2, en el caso en que los componentes de crominancia no se dividan en sub-bloques superior e inferior. Por lo tanto, la división de los componentes de crominancia en sub-bloques superior e inferior da como resultado ocho TU cuadradas, en comparación con cuatro TU rectangulares con una razón de aspecto de 1 a 2 en el caso en que las TU no estén divididas en sub-bloques superior e inferior. Cabe señalar que la CU en la FIG. 8B puede ser una CU de tamaños 4x4, 8x8, 16x16, 32x32 o 64x64. En un ejemplo, el componente de crominancia incluye componentes de crominancia primero y segundo, por ejemplo, superior e inferior. La codificación de los sub-bloques primero y segundo incluye la codificación de los sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la codificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular.

En un procedimiento ejemplar de codificación de datos de vídeo, el codificador de vídeo 20 o el decodificador de vídeo 30 pueden dividir un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, en sub-bloques cuadrados primero y segundo, como se ilustra en la FIG. 8B. El codificador de vídeo 20 o el decodificador de vídeo 30 puede codificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El codificador de vídeo 20 o el decodificador de vídeo 30 también puede decodificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El codificador de vídeo 20 o el decodificador de vídeo 30 generalmente no decodifican un CBF para el bloque de crominancia rectangular. En algunos ejemplos, el primer sub-bloque puede ser un sub-bloque superior y el segundo sub-bloque puede ser un sub-bloque inferior del bloque de crominancia rectangular. El bloque de crominancia rectangular tiene un formato de muestreo 4:2:2.

La FIG. 9 es un diagrama de flujo que ilustra un procedimiento ejemplar para la decodificación de datos de vídeo según los sistemas y procedimientos descritos en este documento. En el procedimiento ejemplar de decodificación de datos de vídeo ilustrado en la FIG. 9, un decodificador, tal como el decodificador de vídeo 30, decodifica datos de vídeo en los cuales los datos de crominancia están dispuestos en un bloque de crominancia rectangular, comprendiendo conjuntamente los sub-bloques cuadrados primero y segundo un bloque de crominancia rectangular (902). Por ejemplo, el módulo de predicción 81 del decodificador de vídeo 30 obtiene un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, que se divide en primeros sub-bloques cuadrados y segundos sub-bloques cuadrados.

En algunos ejemplos, el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular. Además, en algunos ejemplos, el bloque de crominancia rectangular tiene un formato de muestreo de 4:2:2.

El decodificador de vídeo 30 codifica un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero (904). Por ejemplo, el módulo de decodificación por entropía 80 del decodificador de vídeo 30 puede decodificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero.

El decodificador de vídeo 30 también codifica un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero (906), todo ello sin codificar a la vez un CBF para el bloque de crominancia rectangular, es decir, para el bloque global de crominancia rectangular. Por ejemplo, el módulo de decodificación por entropía 80 del decodificador de vídeo 30 puede decodificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero, y puede decodificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero.

En algunos ejemplos, el componente de crominancia incluye componentes de crominancia primero y segundo, en donde la decodificación de los sub-bloques primero y segundo comprende la decodificación de los sub-bloques cuadrados primero y segundo de un primer bloque de crominancia rectangular y la decodificación de los sub-bloques cuadrados primero y segundo de un segundo bloque de crominancia rectangular. El decodificador de vídeo 30, por ejemplo, el módulo de decodificación por entropía 80, puede decodificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. El módulo de decodificación por entropía 80 del decodificador de vídeo 30 también puede decodificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero. Como se describe aquí, en algunos ejemplos, el decodificador de vídeo 30 generalmente no decodifica un CBF para el bloque de crominancia rectangular.

La FIG. 10 es un diagrama de flujo que ilustra un procedimiento ejemplar para la codificación de datos de vídeo según los sistemas y procedimientos descritos en este documento. En el procedimiento ejemplar de codificación de datos de vídeo ilustrado en la FIG. 10, un codificador tal como un codificador de vídeo 20 divide un bloque de

5 crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, en primeros sub-bloques cuadrados y segundos sub-bloques cuadrados (1002). Por ejemplo, el módulo de predicción 41 del codificador de vídeo 20 divide un bloque de crominancia rectangular, por ejemplo, en una hoja de árbol cuádruple, en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado.

10 En algunos ejemplos, el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular. Además, en algunos ejemplos, el primer sub-bloque y el segundo sub-bloque comprenden bloques de crominancia que tienen un formato de muestreo 4:2:2.

15 El codificador de vídeo 20 codifica un primer indicador de bloque codificado (CBF) para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero (1004). Por ejemplo, el módulo de codificación por entropía 56 del codificador de vídeo 20 puede codificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero.

20 El codificador de vídeo 20 también codifica un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero (1006), todo ello sin codificar a la vez un CBF para el bloque de crominancia rectangular. Por ejemplo, el módulo de codificación por entropía 56 del codificador de vídeo 20 puede codificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye o no al menos un coeficiente de transformación distinto de cero.

25 La FIG. 11 es un diagrama de flujo que ilustra un procedimiento ejemplar para la decodificación de datos de vídeo según los sistemas y procedimientos descritos en este documento. En el procedimiento ejemplar de la FIG. 11, un decodificador de vídeo 30 recibe un bloque de crominancia rectangular. El bloque de crominancia rectangular puede ser dividido en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado por el decodificador de vídeo 30 (1102). En un ejemplo, el decodificador de vídeo 30 recibe un bloque de crominancia rectangular, con el bloque de crominancia rectangular dividido en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. El primer sub-bloque cuadrado puede ser un sub-bloque cuadrado superior, y el segundo bloque de transformación cuadrado puede ser un sub-bloque cuadrado inferior, del bloque de crominancia rectangular.

30 El decodificador de vídeo 30 reconstruye un primer bloque de transformación cuadrado (1104). En algunos ejemplos, la reconstrucción del primer bloque de transformación cuadrado puede incluir el uso de un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. En un ejemplo, el decodificador de vídeo 30 intra-decodifica el primer bloque, por ejemplo, el sub-bloque cuadrado superior, de los datos de crominancia intra-predichos, usando un primer bloque de datos de vídeo intra-predichos y los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado.

35 El decodificador de vídeo 30 intra-predice el segundo sub-bloque cuadrado usando muestras de referencia procedentes del primer sub-bloque cuadrado reconstruido (1106). La intra-predicción del segundo sub-bloque cuadrado puede incluir la generación de un segundo bloque de datos de crominancia intra-predichos, utilizando muestras de referencia a partir del primer sub-bloque cuadrado reconstruido, y la reconstrucción del segundo sub-bloque de transformación cuadrado, utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado (1108).

40 El decodificador de vídeo 30 puede reconstruir el segundo sub-bloque cuadrado utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado. Por ejemplo, el decodificador de vídeo 30 puede reconstruir un segundo bloque, por ejemplo, el sub-bloque cuadrado inferior, de datos de crominancia, utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado.

45 En algunos ejemplos, el decodificador de vídeo puede cuantizar de forma inversa los coeficientes de transformación del primer sub-bloque cuadrado, transformar de forma inversa los coeficientes de transformación cuantizados de forma inversa, para generar los primeros datos residuales, y añadir los primeros datos residuales a un primer bloque de datos de crominancia intra-predichos para generar el primer bloque de datos de crominancia. El primer bloque de datos de crominancia intra-predichos para el primer sub-bloque cuadrado se puede generar utilizando muestras de predicción de intra-predicción a partir de uno o más bloques espacialmente contiguos. Las muestras de referencia de intra-predicción del primer sub-bloque se pueden utilizar a continuación para intra-predicir un bloque de datos de crominancia para el segundo sub-bloque cuadrado.

50 En algunos ejemplos, el decodificador de vídeo 30 puede, adicionalmente o alternativamente, reconstruir el segundo bloque de datos de crominancia, por ejemplo, el sub-bloque cuadrado inferior, utilizando muestras de intra-predicción a partir de un bloque contiguo adicional que se suma al bloque de muestras de intra-predicción, proporcionadas por el primer bloque de datos de crominancia, por ejemplo, el sub-bloque cuadrado superior. Algunas modalidades de intra-predicción pueden utilizar muestras de píxeles de un bloque superior espacialmente adyacente (tal como el sub-bloque cuadrado superior en este ejemplo) por encima de un bloque a ser intra-

codificado (tal como el sub-bloque cuadrado inferior en este ejemplo, ya sea solo o en combinación con muestras de píxeles de otro bloque espacialmente contiguo, por ejemplo, en varias intra-modalidades direccionales. En algunos ejemplos, el bloque contiguo adicional comprende un bloque contiguo a la izquierda con relación al bloque a codificar, por ejemplo, adyacente al sub-bloque cuadrado inferior del componente de crominancia.

5 La FIG. 12 es un diagrama de flujo que ilustra un procedimiento ejemplar para la codificación de datos de vídeo según los sistemas y procedimientos descritos en este documento. En el procedimiento ejemplar de la FIG. 12, un codificador de vídeo 20 recibe un bloque de crominancia rectangular. El bloque de crominancia rectangular puede ser dividido en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado por el codificador de vídeo 20 (1202). Por ejemplo, el módulo de predicción 41 del codificador de vídeo 20 puede recibir un bloque de crominancia rectangular, y dividir el bloque de crominancia rectangular en un primer sub-bloque cuadrado y un segundo sub-bloque cuadrado. El primer sub-bloque cuadrado puede ser un sub-bloque cuadrado superior, y el segundo bloque de transformación cuadrado puede ser un sub-bloque cuadrado inferior, del bloque de crominancia rectangular.

15 El codificador de vídeo 20 reconstruye el primer bloque de transformación cuadrado (1204). En algunos ejemplos, la reconstrucción del primer bloque de transformación cuadrado puede incluir el uso de un primer bloque de datos de crominancia intra-predichos y los primeros datos residuales para el primer sub-bloque cuadrado. Por ejemplo, el codificador de vídeo 20 intra-codifica el primer bloque, por ejemplo, el sub-bloque cuadrado superior, y, tras la reconstrucción del primer bloque, utiliza los datos reconstruidos para intra-codificar el segundo bloque, por ejemplo, el sub-bloque cuadrado inferior. En particular, el codificador de vídeo 20 puede codificar el primer bloque mediante la generación de un bloque de datos de crominancia intra-predichos y un bloque de datos residuales que representan una diferencia entre los valores de píxeles de los datos de crominancia intra-predichos y los valores de los píxeles del primer bloque. El codificador de vídeo 20 puede transformar los datos residuales en un bloque de coeficientes de transformación y cuantizar los coeficientes de transformación. El codificador de vídeo 20 puede cuantizar de forma inversa y transformar de forma inversa los coeficientes para reconstruir los datos residuales, y a continuación reconstruir el primer sub-bloque cuadrado de datos de crominancia mediante la generación de un primer bloque de datos de crominancia intra-predichos para el primer sub-bloque cuadrado, y añadiendo los datos de crominancia intra-predichos a los primeros datos residuales obtenidos a partir del primer sub-bloque cuadrado. El codificador de vídeo 20 también codifica por entropía los datos residuales y la información de la modalidad de intra-codificación, para su transmisión al decodificador de vídeo 30 en un flujo de bits de vídeo codificado.

Para intra-codificar el segundo sub-bloque cuadrado (por ejemplo, el sub-bloque cuadrado inferior), el codificador de vídeo 20 utiliza muestras de referencia de intra-predicción del primer sub-bloque cuadrado, para generar un bloque de datos de crominancia intra-predichos, y a continuación genera datos residuales que indican la diferencia entre los datos de crominancia intra-predichos y los datos de crominancia del segundo sub-bloque cuadrado. A continuación, el codificador de vídeo 20 codifica por entropía los datos residuales y la información de modalidad, e incluye los datos codificados por entropía en el flujo de bits de vídeo codificado para su uso por el decodificador de vídeo 30 en operaciones de decodificación. De esta manera, los datos de crominancia del sub-bloque cuadrado superior reconstruido se pueden utilizar como muestras de referencia de intra-predicción para intra-codificar el sub-bloque cuadrado inferior de datos de crominancia.

El codificador de vídeo 20 intra-predice el segundo sub-bloque cuadrado usando muestras de referencia del primer sub-bloque cuadrado reconstruido (1206). La intra-predicción del segundo sub-bloque cuadrado puede incluir la generación de un segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales para el segundo sub-bloque cuadrado, en base a muestras de intra-predicción en el primer sub-bloque cuadrado; y la codificación del segundo sub-bloque de transformación cuadrado utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales (1208).

El codificador de vídeo 20 puede reconstruir un segundo bloque, por ejemplo, el sub-bloque cuadrado inferior, de datos de crominancia utilizando el segundo bloque de datos de crominancia intra-predichos y los segundos datos residuales del segundo sub-bloque cuadrado. En algunos ejemplos, el codificador de vídeo, por ejemplo, en el bucle de reconstrucción del codificador de vídeo 20, puede cuantizar de forma inversa coeficientes de transformación del primer sub-bloque cuadrado, transformar de forma inversa los coeficientes de transformación cuantizados de forma inversa, para generar los primeros datos residuales, y añadir los primeros datos residuales a un primer bloque de datos de crominancia intra-predichos para generar el primer bloque de datos de crominancia. El primer bloque de datos de crominancia intra-predichos para el primer sub-bloque cuadrado se puede generar utilizando muestras de predicción de intra-predicción a partir de uno o más bloques espacialmente contiguos. Las muestras de referencia de intra-predicción del primer sub-bloque se pueden utilizar a continuación para intra-predicir un bloque de datos de crominancia para el segundo sub-bloque cuadrado.

En algunos ejemplos, el codificador de vídeo 20 puede, adicionalmente o alternativamente, reconstruir el segundo bloque de datos de crominancia, por ejemplo, el sub-bloque cuadrado inferior, utilizando muestras de intra-predicción de un bloque contiguo adicional, que se suma al bloque de muestras de intra-predicción proporcionado por el primer bloque de datos de crominancia, por ejemplo, el sub-bloque cuadrado superior. Algunas modalidades de intra-predicción pueden utilizar muestras de píxeles de un bloque superior espacialmente adyacente (tal como el sub-bloque cuadrado superior en este ejemplo) por encima de un bloque a ser intra-codificado (tal como el sub-

bloque cuadrado inferior en este ejemplo, ya sea solo o en combinación con muestras de píxeles de otro bloque espacialmente contiguo, por ejemplo, en varias intra-modalidades direccionales. En algunos ejemplos, el bloque contiguo adicional comprende un bloque contiguo a la izquierda, con relación al bloque a codificar, por ejemplo, adyacente al sub-bloque cuadrado inferior del componente de crominancia.

5 Algunos ejemplos incluyen sistemas de codificación que proveen el uso de transformaciones cuadradas incluso en el caso de vídeo en formato 4:2:2. La estructura básica de árbol cuádruple de la HEVC puede mantenerse de manera que cada bloque rectangular de un componente de crominancia de 4:2:2 pueda dividirse adicionalmente en 4 bloques rectangulares. Por ejemplo, un bloque de crominancia de tamaño 16x32 puede dividirse en 4 bloques de tamaño 8x16. Cada uno de los bloques de tamaño 8x16 puede subdividirse además en 4 bloques de tamaño 4x8. En la hoja del árbol cuádruple, es decir, cuando el bloque ya no se divide más, el bloque rectangular para un componente de crominancia de 4:2:2 se divide en 2 bloques cuadrados y se aplica una transformación cuadrada. Por ejemplo, se utilizan 2 transformaciones de tamaño 8x8 para un bloque de hoja de tamaño 8x16. El proceso de selección del QP puede ser igual al del perfil principal de la HEVC. No se realiza ningún ajuste al QP en  $\pm 3$ . El bloque rectangular dividido de, por ejemplo, un componente de crominancia de 4:2:2, puede recibirse, por ejemplo, en un decodificador de vídeo. En consecuencia, tanto un codificador de vídeo como un decodificador de vídeo pueden funcionar en los primeros sub-bloques cuadrados y los segundos sub-bloques cuadrados. El codificador de vídeo puede dividir un bloque de crominancia rectangular en los sub-bloques cuadrados primero y segundo. El decodificador de vídeo puede recibir estos sub-bloques cuadrados primero y segundo.

20 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, como una o más instrucciones o código, pueden almacenarse en, o transmitirse por, 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 desde un lugar a otro, por ejemplo, según un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, generalmente, a (1) medios de almacenamiento tangibles y legibles por ordenador, que sean no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser medios disponibles cualesquiera, a los que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

35 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 adecuadamente un medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una sede de la Red, 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, debería 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 orientan a medios de almacenamiento tangibles no transitorios. Los discos, como se usan en el presente documento, incluyen el disco compacto (CD), el disco de láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos normalmente reproducen datos de manera magnética, mientras que otros discos reproducen los datos de manera óptica con láser. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

50 Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones lógicas programables en el terreno (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. 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 dentro de hardware especializado y/o módulos de software configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

65 Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un equipo de mano 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 divulgadas, pero no requieren necesariamente la realización mediante diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, pueden

combinarse diversas unidades en una unidad de hardware de códec, o ser proporcionadas por una colección de unidades de hardware inter-operativas, incluyendo uno o más procesadores, como se ha descrito anteriormente, junto con el software y/o firmware adecuado.

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

## REIVINDICACIONES

1. Un procedimiento de decodificación de datos de vídeo, procedimiento que comprende:
  - 5 en base a un formato de muestreo de crominancia de una unidad de codificación, la obtención (902), para un nodo hoja de la unidad de codificación, de un bloque de crominancia rectangular comprendiendo sub-bloques cuadrados primero y segundo;
  - 10 en base al nodo hoja comprendiendo el bloque de crominancia rectangular, la no decodificación de un indicador de bloque codificado, CBF, para todo el bloque de crominancia rectangular;
  - la decodificación (904) de un primer CBF para el primer sub-bloque cuadrado, indicativo de si el primer sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero; y
  - 15 la decodificación (906) de un segundo CBF para el segundo sub-bloque cuadrado, indicativo de si el segundo sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero.
2. El procedimiento de la reivindicación 1, en el que el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular y, preferiblemente, en el que la unidad de codificación tiene un formato de muestreo 4:2:2, y aún más preferiblemente, que comprende además:
  - 25 la generación de datos residuales usando los coeficientes de transformación; y la reconstrucción de los datos de vídeo codificados utilizando los datos residuales.
3. Un procedimiento de codificación de datos de vídeo, procedimiento que comprende:
  - 30 en base a un formato de muestreo de crominancia de una unidad de codificación, la división de un bloque de crominancia rectangular correspondiente al nodo hoja en sub-bloques cuadrados primero y segundo;
  - en base al nodo hoja comprendiendo el bloque de crominancia rectangular, la no codificación de un indicador de bloque codificado, CBF, para todo el bloque de crominancia rectangular;
  - 35 la codificación de un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero; y
  - la codificación de un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero.
4. El procedimiento de la reivindicación 3, en el que el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular.
5. El procedimiento de las reivindicaciones 3 o 4, en el que la unidad de codificación tiene un formato de muestreo 4:2:2.
6. El procedimiento de la reivindicación 3, que comprende además:
  - 50 generar datos residuales usando los coeficientes de transformación; y codificar datos de vídeo utilizando los datos residuales.
7. Un aparato para la decodificación (14) de datos de vídeo, que comprende:
  - 55 uno o más procesadores configurados para:
    - en base a un formato de muestreo de crominancia de una unidad de codificación, obtener, para un nodo hoja de la unidad de codificación, un bloque de crominancia rectangular que comprende sub-bloques cuadrados primero y segundo;
    - 60 en base al nodo hoja comprendiendo el bloque de crominancia rectangular, no decodificar un indicador de bloque codificado, CBF, para todo el bloque de crominancia rectangular;
    - decodificar un primer CBF para el primer sub-bloque cuadrado, indicativo de si el primer sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero; y
    - 65 decodificar un segundo CBF para el segundo sub-bloque cuadrado, indicativo de si el segundo sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero.

8. El aparato de la reivindicación 7, en el que el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular.
- 5 9. El aparato de las reivindicaciones 7 u 8, en el que la unidad de codificación tiene un formato de muestreo 4:2:2.
10. El aparato de la reivindicación 7, estando los uno o más procesadores configurados adicionalmente para:
- 10 generar datos residuales utilizando los coeficientes de transformación; y reconstruir los datos de vídeo codificado utilizando los datos residuales.
11. Un aparato para la codificación (12) de datos de vídeo, que comprende:
- 15 uno o más procesadores configurados para:
- en base a un formato de muestreo de crominancia de una unidad de codificación, dividir un bloque de crominancia rectangular, correspondiente a un nodo hoja de la unidad de codificación, en sub-bloques cuadrados primero y segundo;
- 20 en base al nodo hoja comprendiendo el bloque de crominancia rectangular, no codificar un indicador de bloque codificado, CBF, para todo el bloque de crominancia rectangular;
- codificar un primer CBF para el primer sub-bloque cuadrado, para indicar si el primer sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero;
- 25 y codificar un segundo CBF para el segundo sub-bloque cuadrado, para indicar si el segundo sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero.
- 30 12. El aparato de la reivindicación 11, en el que el primer sub-bloque comprende un sub-bloque superior y el segundo sub-bloque comprende un sub-bloque inferior del bloque de crominancia rectangular.
13. El aparato de las reivindicaciones 11 o 12, en el que la unidad de codificación tiene un formato de muestreo 4:2:2.
- 35 14. El aparato de la reivindicación 11, en el que los uno o más procesadores están configurados adicionalmente para:
- generar datos residuales utilizando los coeficientes de transformación; y codificar datos de vídeo utilizando los datos residuales.
- 40 15. Un medio de almacenamiento no transitorio legible por ordenador, que almacena instrucciones que, al ser ejecutadas por uno o más procesadores, hacen que los uno o más procesadores:
- 45 en base a un formato de muestreo de crominancia de una unidad de codificación, obtengan, para un nodo de hoja de la unidad de codificación, un bloque de crominancia rectangular que comprende sub-bloques cuadrados primero y segundo;
- en base al nodo hoja comprendiendo el bloque de crominancia rectangular, no decodifiquen un indicador de bloque codificado, CBF, para todo el bloque de crominancia rectangular;
- 50 codifiquen un primer CBF para el primer sub-bloque cuadrado, indicativo de si el primer sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero; y
- 55 codifiquen un segundo CBF para el segundo sub-bloque cuadrado, indicativo de si el segundo sub-bloque cuadrado incluye al menos un coeficiente de transformación distinto de cero.

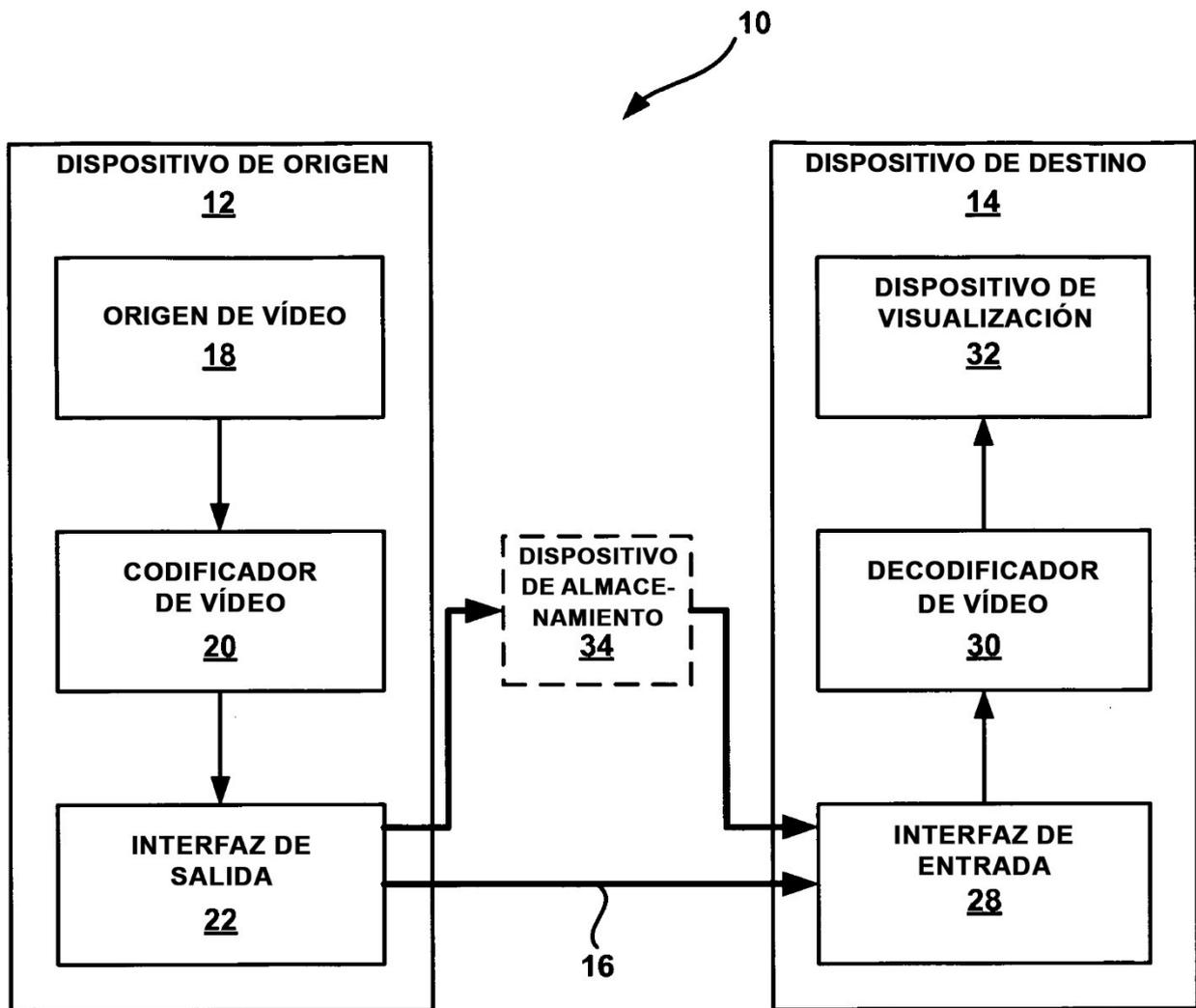


FIG. 1

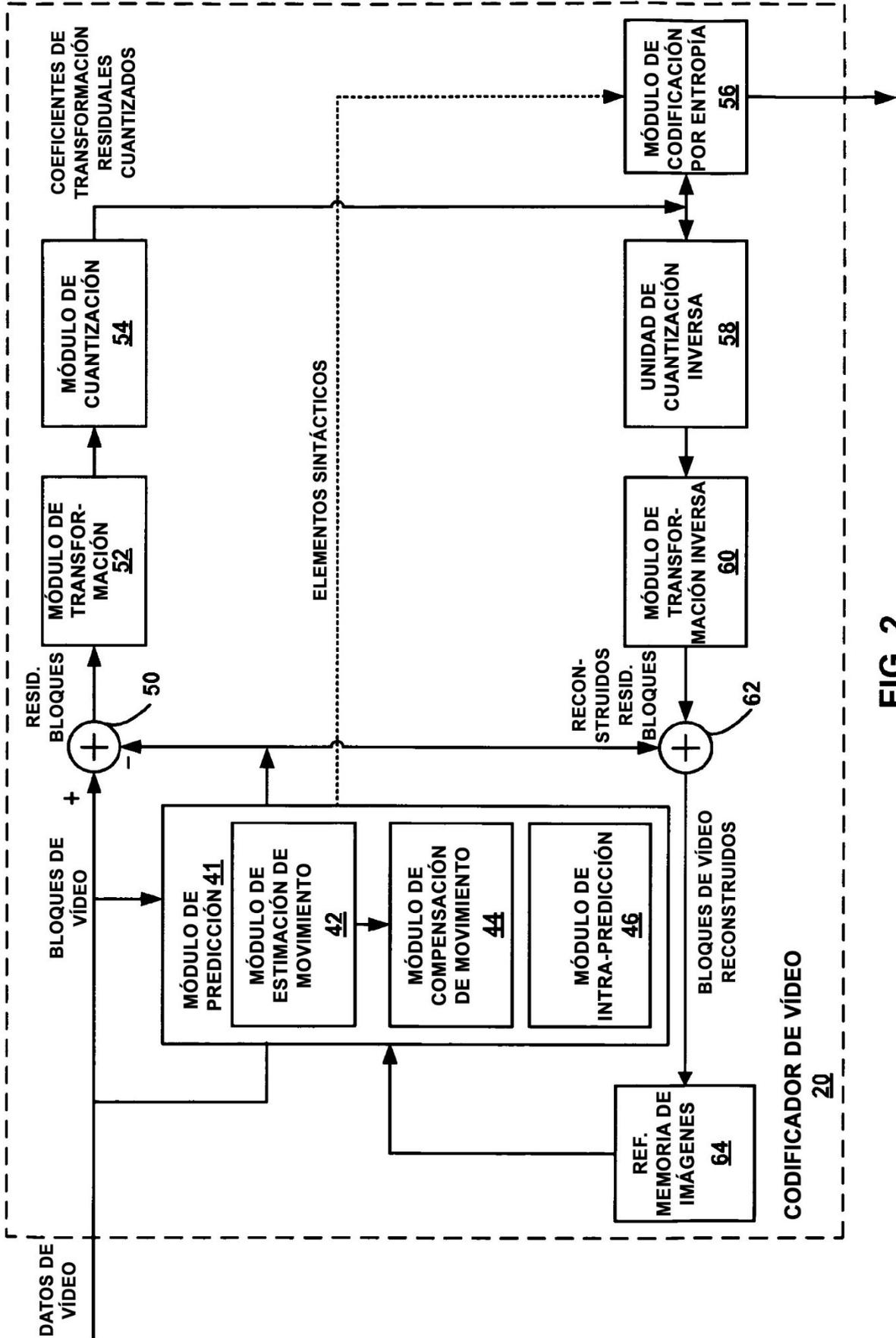


FIG. 2

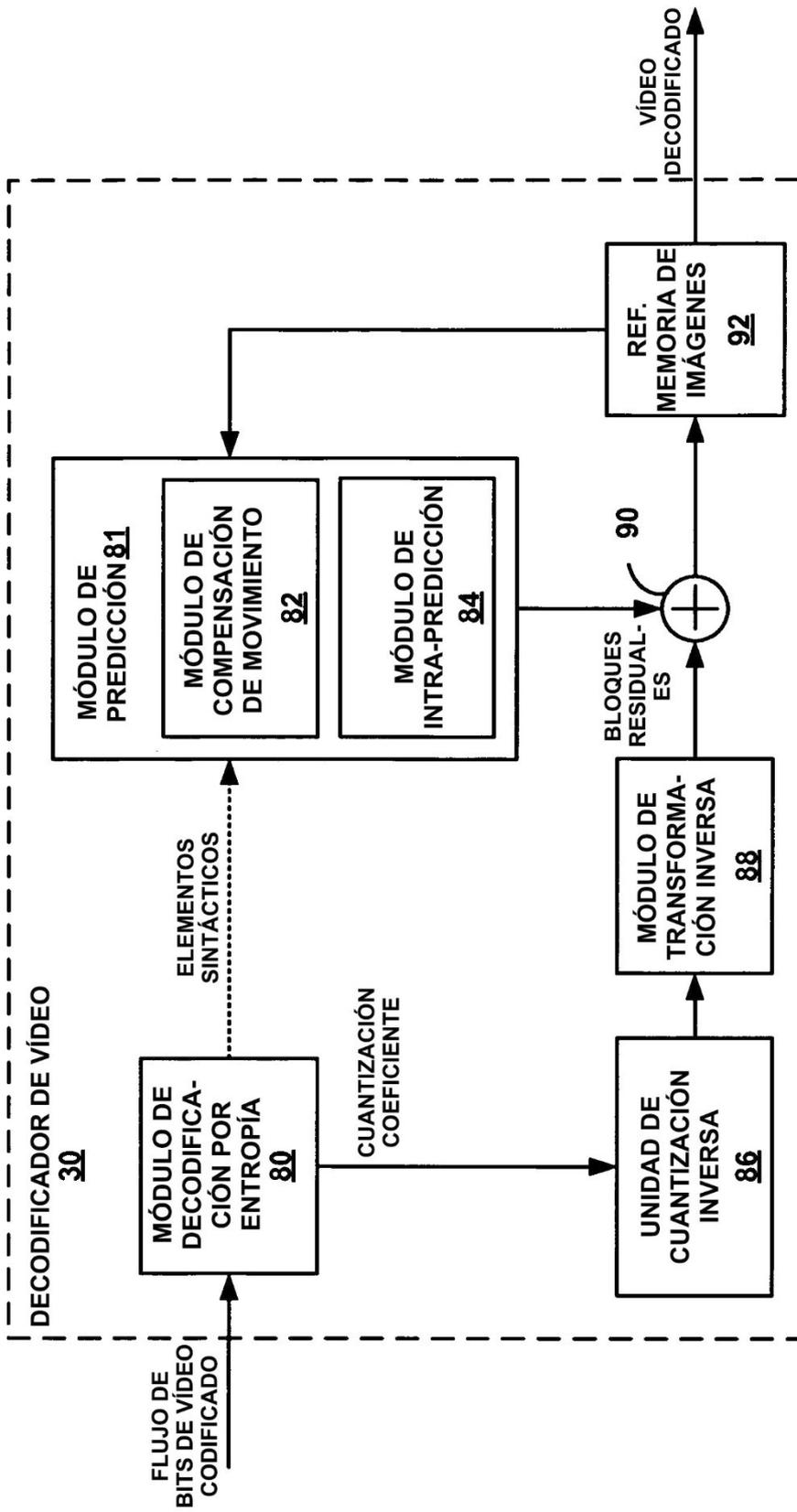
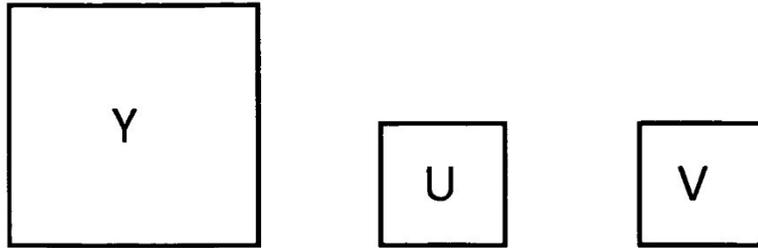
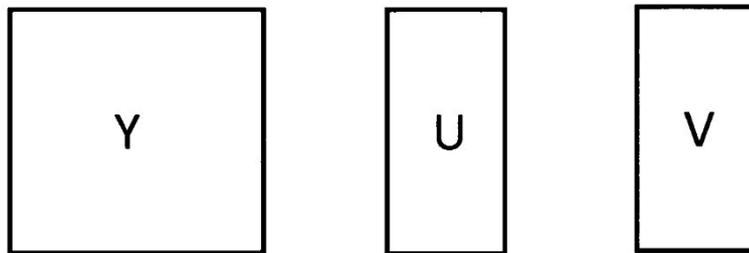


FIG. 3



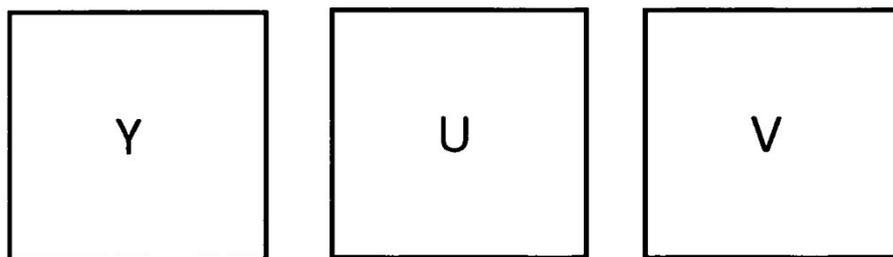
**Formato de muestra 4:2:0**

**FIG. 4A**



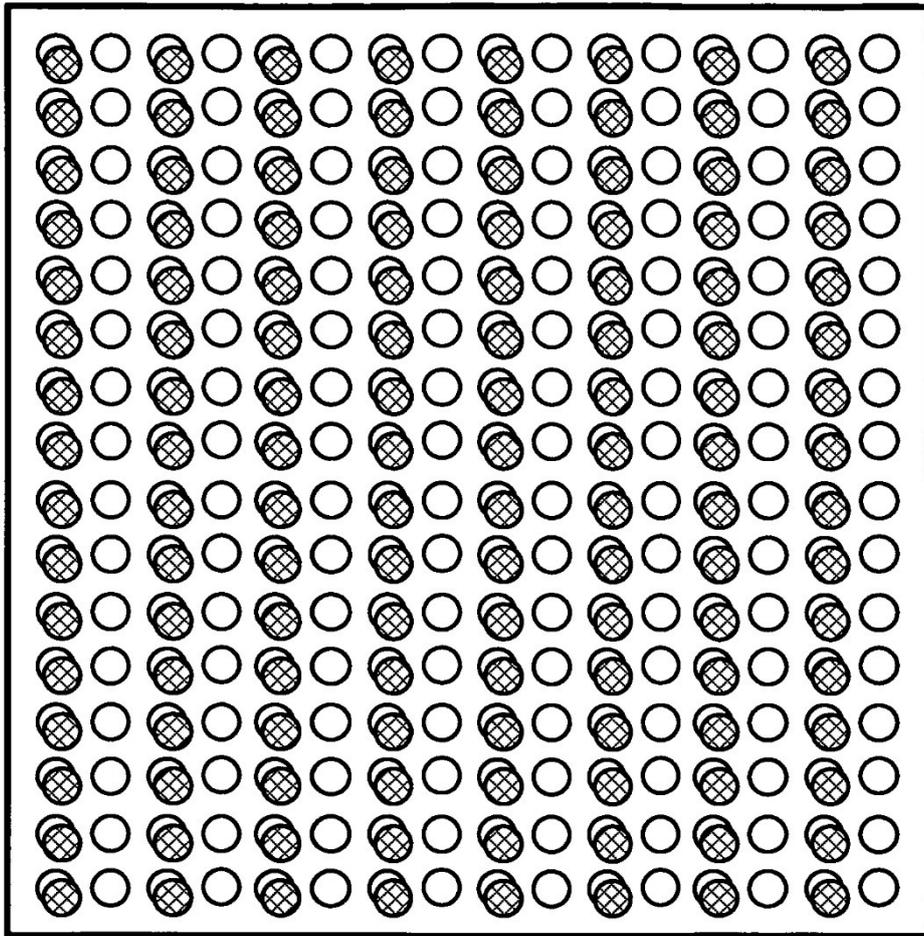
**Formato de muestra 4:2:2**

**FIG. 4B**



**Formato de muestra 4:4:4**

**FIG. 4C**



**CU de tamaño 16x16 con formato de muestra 4:2:2**

- Muestra de luminancia
- ⊗ Muestras de crominancia

**FIG. 5**



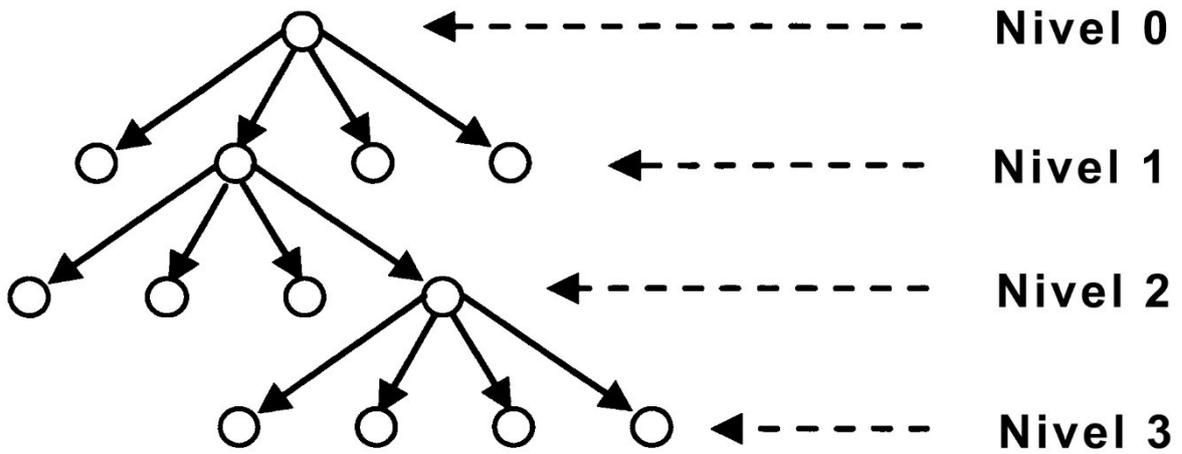
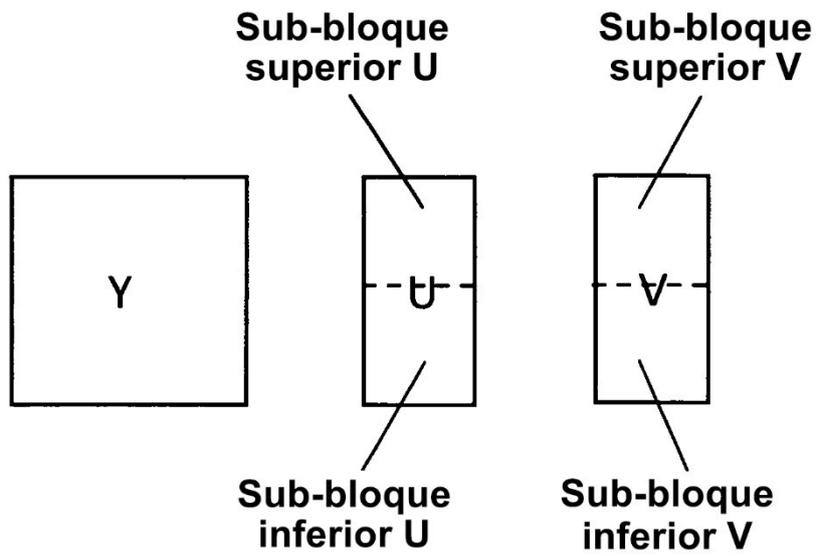


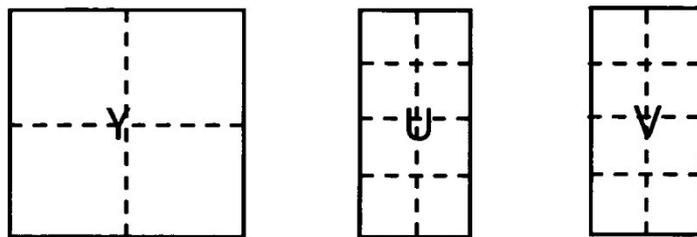
Diagrama de nivel de descomposición en árbol cuádruple

FIG. 7



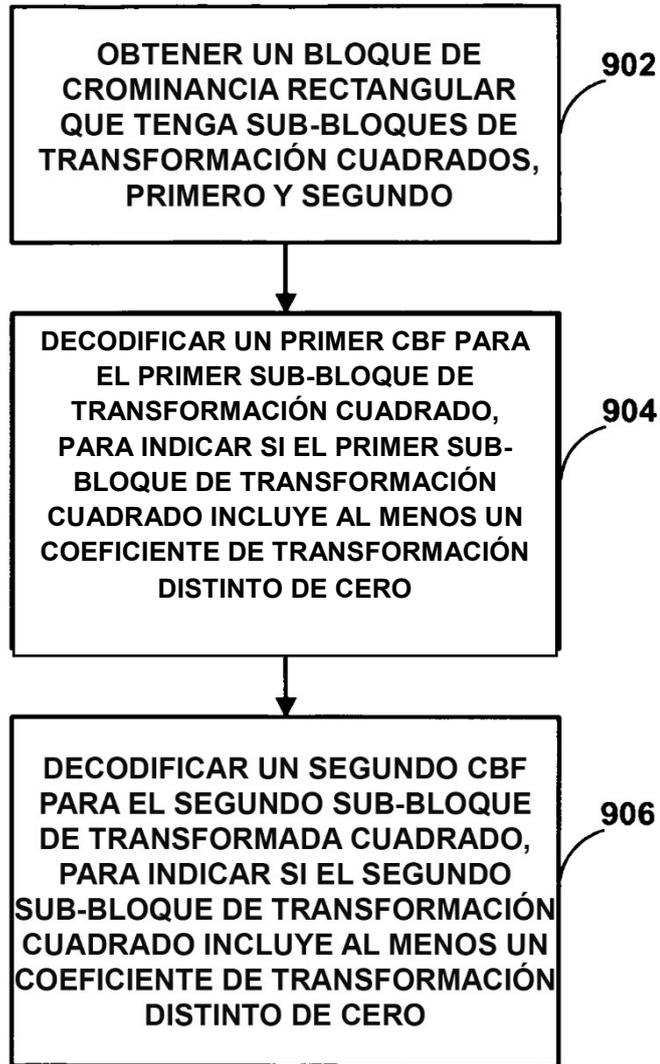
Caso de bloque no dividido

**FIG. 8A**

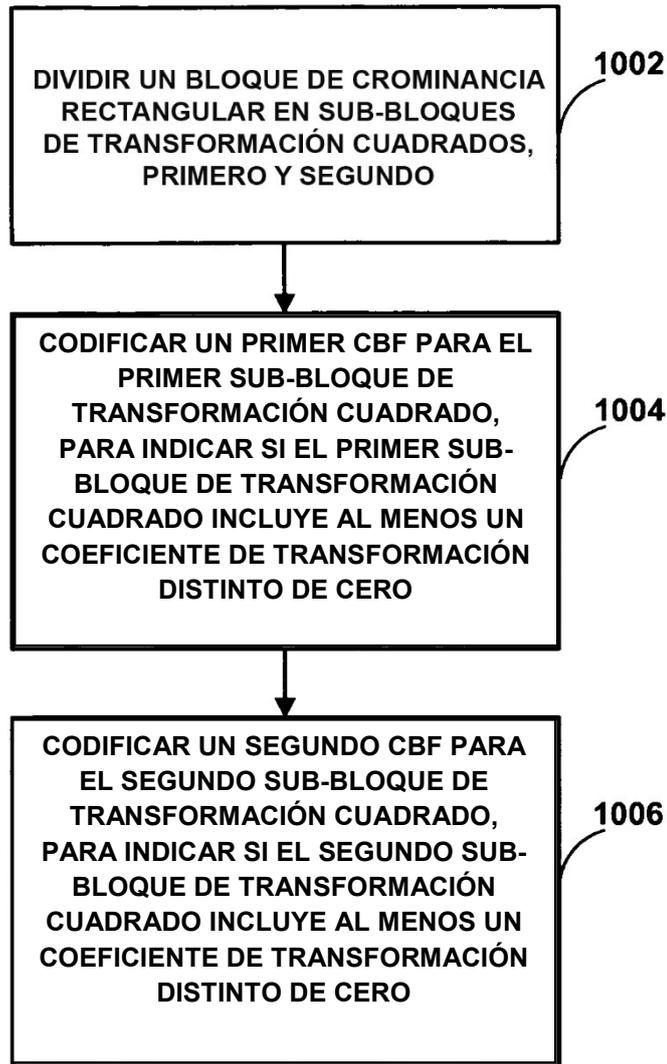


Partición de TU cuadrada

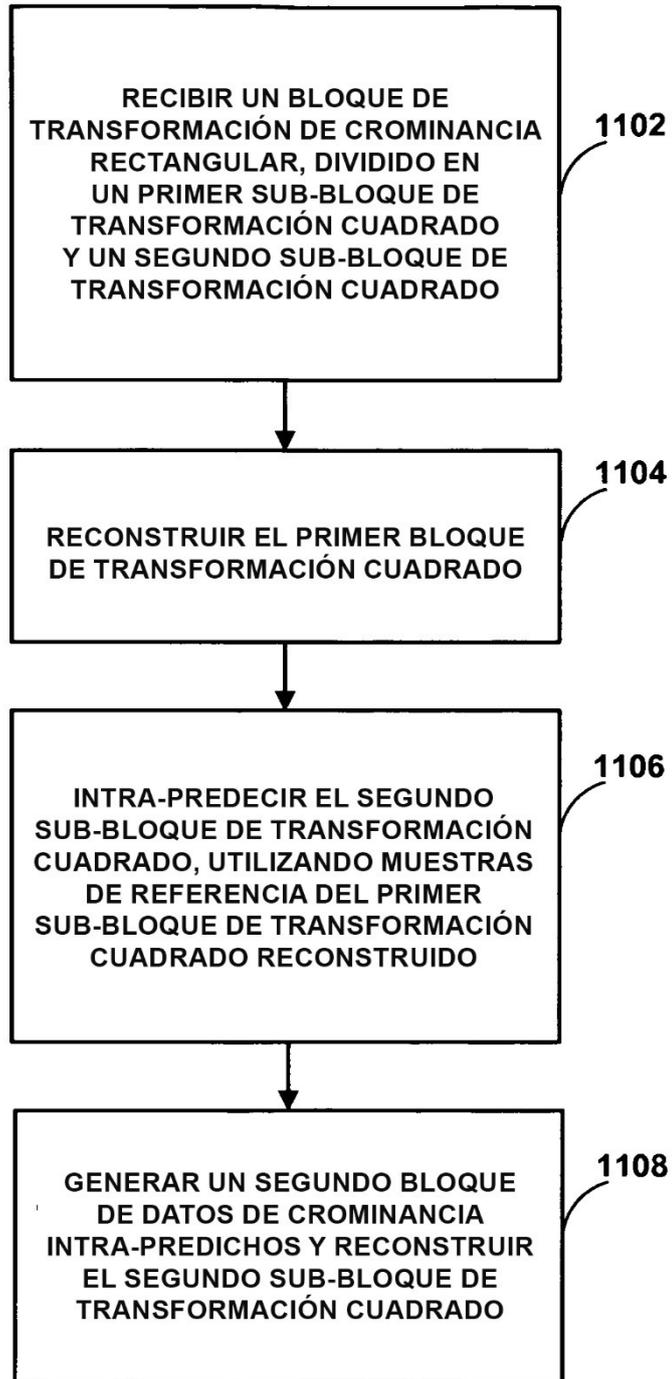
**FIG. 8B**



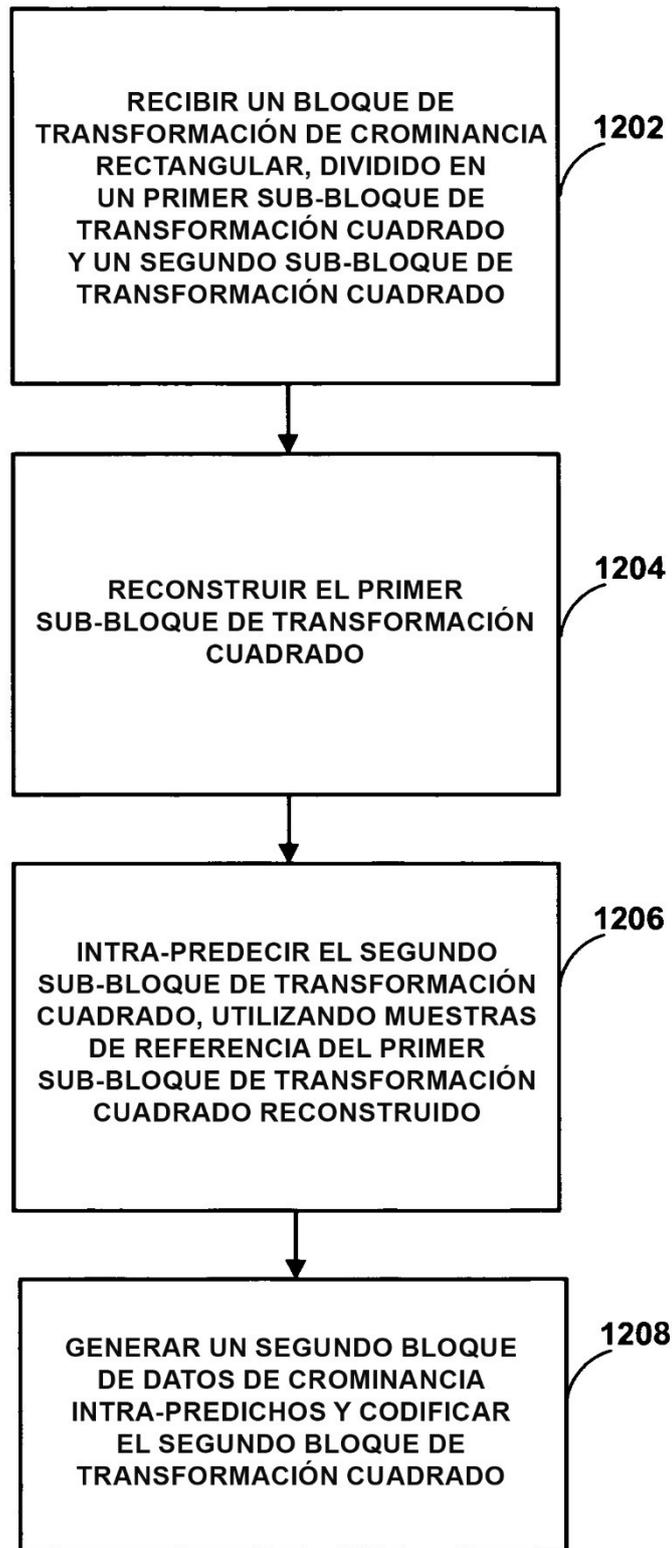
**FIG. 9**



**FIG. 10**



**FIG. 11**



**FIG. 12**