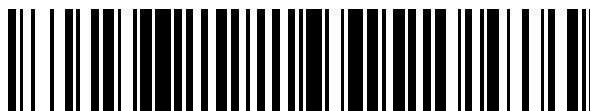


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 702 950**

51 Int. Cl.:

H04N 19/176 (2014.01)
H04N 19/70 (2014.01)
H04N 19/126 (2014.01)
H04N 19/14 (2014.01)
H04N 19/186 (2014.01)
H04N 19/157 (2014.01)
H04N 19/18 (2014.01)
H04N 19/102 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **13.03.2015 PCT/US2015/020529**
 87 Fecha y número de publicación internacional: **17.09.2015 WO15138954**
 96 Fecha de presentación y número de la solicitud europea: **13.03.2015 E 15717691 (8)**
 97 Fecha y número de publicación de la concesión europea: **26.09.2018 EP 3117614**

54 Título: **Transformación inversa de color-espacio para vídeo codificado con pérdidas y sin pérdidas**

30 Prioridad:

14.03.2014 US 201461953573 P
18.04.2014 US 201461981645 P
10.10.2014 US 201462062637 P
12.03.2015 US 201514656532

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
06.03.2019

73 Titular/es:

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

72 Inventor/es:

ZHANG, LI;
CHEN, JIANLE y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 702 950 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

Transformación inversa de color-espacio para vídeo codificado con pérdidas y sin pérdidas

5 **[0001]** Esta solicitud reivindica el beneficio de la Solicitud Provisional estadounidense N.º 61/953,573, presentada el viernes 14 de marzo de 2014, de la Solicitud Provisional estadounidense N.º 61/981,645, presentada el viernes 18 de abril de 2014 y de la Solicitud Provisional estadounidense N.º 62/062,637, presentada el viernes 10 de octubre de 2014.

10 **CAMPO TÉCNICO**

[0002] Esta divulgación se refiere a la codificación de vídeo y, más específicamente, a la codificación de vídeo utilizando la conversión de color-espacio.

15 **ANTECEDENTES**

[0003] Las capacidades de 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 continua de vídeo y similares. Los dispositivos de vídeo digital implementan técnicas de codificación 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), la norma de Codificación de Vídeo de Alta Eficacia (HEVC), actualmente en desarrollo, y las ampliaciones 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 codificación de vídeo.

30 **[0004]** Las técnicas de codificación de vídeo incluyen la predicción espacial (intraimagen) 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 intracodificado (I) de una imagen se codifican mediante predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos 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 **[0005]** La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se va a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con una modalidad de intracodificación y los datos residuales. Para una compresión adicional, los datos residuales pueden transformarse desde el dominio del píxel hasta un dominio de transformación, dando como resultado coeficientes de transformación residuales, que a continuación se pueden cuantizar. Los coeficientes de transformación cuantizados, dispuestos inicialmente en una formación bidimensional, pueden examinarse con el fin de generar un vector unidimensional de coeficientes de transformación, y puede aplicarse codificación por entropía para lograr aún más compresión.

50 **[0006]** Las técnicas de codificación y decodificación de vídeo generalmente prevén que los datos de vídeo experimenten una transformación de color-espacio en la etapa de codificación y una transformación inversa de color-espacio en la etapa de decodificación (es decir, en la decodificación local que tiene lugar dentro del bucle de codificación). Generalmente se usan diferentes transformaciones de color-espacio, en función de si los datos de vídeo están codificados en una modalidad con pérdidas o sin pérdidas.

60 **[0007]** El documento US 2005/259730 prevé, en cambio, utilizar la misma transformación de color-espacio y la transformación inversa, es decir, YCoCg-R (una transformación reversible que generalmente se usa en el caso de codificación sin pérdidas), independientemente de si el vídeo está codificado en una modalidad con pérdidas o sin pérdidas. El documento también divulga que para que esto funcione es necesario que los datos codificados con pérdidas experimenten una extensión de bit en un bit (es decir, un desplazamiento de bits hacia la izquierda en 1 bit) antes de aplicar la transformación inversa de color-espacio YCoCg-R.

65 **SUMARIO**

[0008] En general, esta divulgación describe técnicas para codificar bloques de vídeo usando un proceso de

conversión de color-espacio. Un codificador de vídeo, como un codificador de vídeo o un decodificador de vídeo, puede determinar una modalidad de codificación usada para codificar los datos de vídeo. La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. El codificador de vídeo puede determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. El codificador de vídeo puede aplicar el proceso de transformación de color-espacio en la codificación de los datos de vídeo. Al decodificar los datos de vídeo, independientemente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas, el codificador de vídeo puede aplicar el mismo proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación.

[0009] En un ejemplo, se describe un procedimiento para decodificar datos de vídeo. El procedimiento comprende recibir un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio. El procedimiento también incluye recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. El procedimiento incluye además aplicar un proceso de transformación inversa de color-espacio al primer bloque codificado y aplicar el mismo proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo.

[0010] En otro ejemplo, un dispositivo incluye una memoria configurada para almacenar datos de vídeo y uno o más procesadores configurados para recibir un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio. Además, los uno o más procesadores también están configurados para recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. Los uno o más procesadores están configurados además para aplicar un proceso de transformación inversa de color-espacio al primer bloque codificado y aplicar el mismo proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo.

[0011] En otro ejemplo, un dispositivo incluye medios para recibir un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado de datos de vídeo fue codificado utilizando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio, y medios para recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. El dispositivo incluye además medios para aplicar un proceso de transformación inversa de color-espacio al primer bloque codificado de datos de vídeo y medios para aplicar el proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo.

[0012] En otro ejemplo, un medio de almacenamiento legible por ordenador que comprende instrucciones que, cuando se ejecutan, hacen que uno o más procesadores reciban un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado de datos de vídeo fue codificado utilizando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio, y recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. Las instrucciones, cuando se ejecutan, también hacen que el procesador aplique un proceso de transformación inversa de color-espacio al primer bloque codificado de datos de vídeo y aplique el proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo.

[0013] En otro ejemplo, se describe un procedimiento para codificar datos de vídeo. El procedimiento comprende determinar una modalidad de codificación usada para codificar los datos de vídeo, en donde la modalidad de codificación es una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas, y determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación usada para codificar los datos de vídeo. El procedimiento incluye además la aplicación del proceso de transformación de color-espacio a los datos de vídeo. El procedimiento también incluye aplicar un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación, en el que el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

[0014] En otro ejemplo, un dispositivo incluye una memoria configurada para almacenar datos de vídeo y uno o más procesadores configurados para determinar una modalidad de codificación usada para codificar los datos de vídeo, en donde la modalidad de codificación es una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas, y determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. Los uno o más procesadores están configurados además para aplicar el proceso de transformación de color-espacio a los datos de vídeo. Los uno o más procesadores también están configurados para aplicar un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación, en donde el proceso de transformación inversa de color-espacio es independiente de si

la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

5 [0015] En otro ejemplo, un dispositivo incluye medios para determinar una modalidad de codificación usada para codificar los datos de vídeo, en donde la modalidad de codificación es una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas, y medios para determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. El dispositivo incluye además medios para aplicar el proceso de transformación de color-espacio a los datos de vídeo. El dispositivo también incluye medios para aplicar un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación, en el que el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

15 [0016] En otro ejemplo, un medio de almacenamiento legible por ordenador comprende instrucciones que, cuando se ejecutan, hacen que uno o más procesadores determinen una modalidad de codificación usada para codificar los datos de vídeo, en donde la modalidad de codificación es una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas, y determinen un proceso de transformación de color-espacio que dependa de la modalidad de codificación utilizada para codificar los datos de vídeo. Las instrucciones además hacen que los uno o más procesadores apliquen el proceso de transformación de color-espacio a los datos de vídeo. Las instrucciones también hacen que los uno o más procesadores apliquen un proceso de transformación inversa de color-espacio en un ciclo de decodificación del proceso de codificación, en donde el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

25 [0017] Los detalles de uno o más ejemplos de la divulgación se exponen en los dibujos adjuntos y la descripción siguiente. Otras características, objetos y ventajas de la divulgación serán evidentes a partir de la descripción y dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

30 [0018]

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

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

40 La figura 3 es un diagrama de bloques que ilustra un decodificador de vídeo a modo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

La figura 4 es un diagrama conceptual que ilustra las 35 modalidades de predicción de la HEVC de acuerdo con una o más técnicas de la divulgación actual.

45 La figura 5 es un diagrama conceptual que ilustra los candidatos de vectores de movimiento adyacentes espaciales para las modalidades de fusión y de predicción avanzada de vectores de movimiento (AMVP), de acuerdo con una o más técnicas de la divulgación actual.

50 La figura 6 es un diagrama conceptual que ilustra un ejemplo de copia intrabloque (BC) de acuerdo con una o más técnicas de la divulgación actual.

La figura 7 es un diagrama conceptual que ilustra un ejemplo de un bloque de destino y una muestra de referencia para un intrabloque de tamaño 8x8, de acuerdo con una o más técnicas de la divulgación actual.

55 La figura 8 es un diagrama de flujo que ilustra una técnica de codificación de acuerdo con una o más técnicas de la presente divulgación.

La figura 9 es un diagrama de flujo que ilustra una técnica de decodificación según una o más técnicas de la divulgación actual.

60 **DESCRIPCIÓN DETALLADA**

[0019] En algunos ejemplos, esta divulgación está relacionada con la codificación del contenido de la pantalla, en la que se utiliza el formato de muestreo 4: 4: 4 de alta croma. En algunos ejemplos, esta divulgación también es aplicable para extensiones de rango (RCEx), incluido el soporte del formato 4:4:4 de muestreo de alta croma, de profundidad de bits posiblemente alta (más de 8 bits). En algunos ejemplos, esta divulgación también es aplicable a otros formatos de color, tales como el formato de muestreo de croma 4: 2: 2. Más específicamente, en esta divulgación se describen

muchas técnicas diferentes relacionadas con la conversión de color-espacio.

[0020] Entre las normas de codificación de vídeo se incluyen ITU-T H.261, Visual del MPEG-1 de la ISO/IEC, ITU-T H.262 o Visual del MPEG-2 de la ISO/IEC, ITU-T H.263, Visual del MPEG-4 de la ISO/IEC e ITU-T H.264 (también conocida como AVC del MPEG-4 de la ISO/IEC), incluyendo sus ampliaciones de codificación de vídeo ajustable a escala (SVC) y de codificación de vídeo de múltiples vistas (MVC).

[0021] Ha finalizado el diseño de una nueva norma de codificación de vídeo, concretamente la Codificación de Vídeo de Alta Eficacia (HEVC), por parte del Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de la ITU-T y del Grupo de Expertos en Imágenes en Movimiento (MPEG) de la ISO/IEC.

[0022] En la HEVC, la unidad de codificación más grande en un fragmento se denomina bloque arbolado de codificación (CTB). Un CTB contiene un árbol cuádruple cuyos nodos son unidades de codificación. El tamaño de un CTB puede variar de 16x16 a 64x64 en el perfil principal de la HEVC, aunque se puede prestar soporte a CTB de tamaños de 8x8. Una unidad de codificación (CU) podría ser del mismo tamaño que un CTB y tan pequeña como de tamaño 8x8. Cada CU está codificada con una modalidad. Cuando una CU está intercodificada, la CU puede dividirse adicionalmente en dos unidades de predicción (PU) o convertirse en una única PU cuando no se aplican particiones adicionales. Cuando dos PU están presentes en una CU, cada PU puede ser rectángulos de mitad de tamaño o dos rectángulos con un tamaño igual a 1/4 o 3/4 del tamaño de la CU.

[0023] Cuando la CU está intercodificada, está presente un conjunto de información de movimiento para cada PU. Además, cada PU se codifica con una modalidad única de interpredicción para obtener el conjunto de información de movimiento. En la HEVC, los tamaños de PU más pequeños son 8x4 y 4x8.

[0024] La HEVC especifica cuatro unidades de transformación (TU) con tamaños de 4x4, 8x8, 16x16 y 32x32 para codificar el residuo de predicción. Un CTB puede dividirse recursivamente en 4 o más TU. Las TU utilizan funciones basadas en enteros que son similares a las funciones de la transformación de coseno discreta (DCT). Además, los bloques de transformación de luma de tamaño 4x4 que pertenecen a una región intracodificada se transforman utilizando una transformación entera obtenida de una función de transformación de seno discreta (DST). La cromina utiliza los mismos tamaños de TU que la luma.

[0025] En la norma actual de HEVC, para el componente de luma de cada Unidad de Predicción (PU), se utiliza un procedimiento de intrapredicción con 33 modalidades de predicción angular (indizadas de 2 a 34), una modalidad DC (indizada con 1) y una modalidad Plana (indizada con 0), como se describe a continuación con respecto a la figura 4.

[0026] Además de las 35 intramodalidades anteriores, la HEVC también emplea una modalidad más, denominada 'I-PCM'. En la modalidad I-PCM, la predicción, la transformación, la cuantización y la codificación por entropía se omiten mientras que las muestras de predicción sean codificadas mediante un número predefinido de bits. El propósito principal de la modalidad I-PCM es gestionar la situación cuando la señal no puede ser codificada eficazmente por otras modalidades.

[0027] En la norma de HEVC actual, hay dos modalidades de interpredicción disponibles. Una de estas modalidades es la modalidad de fusión (el salto se considera como un caso especial de la fusión), y la segunda es la modalidad de predicción avanzada de vector de movimiento (AMVP) para una unidad de predicción (PU).

[0028] Tanto en la AMVP como en la modalidad de fusión, se mantiene una lista de candidatos de vectores de movimiento (MV) para múltiples predictores de vectores de movimiento. El (los) vector (es) de movimiento, así como los índices de referencia en la modalidad de fusión, de la PU actual se generan tomando un candidato de la lista de candidatos de MV.

[0029] La lista de candidatos de MV puede contener hasta 5 candidatos para la modalidad de fusión y solo dos candidatos para la modalidad de AMVP. Un candidato de fusión puede contener un conjunto de información de movimiento, por ejemplo, vectores de movimiento correspondientes a ambas listas de imágenes de referencia (lista 0 y lista 1) y los índices de referencia. Si un candidato de fusión se identifica mediante un índice de fusión, las imágenes de referencia se utilizan para la predicción de los bloques actuales, y también se determinan los vectores de movimiento asociados. Sin embargo, en la modalidad de AMVP, para cada dirección de predicción potencial de la lista 0 o la lista 1, un índice de referencia necesita ser señalado explícitamente, junto con un índice de MVP para la lista de candidatos de MV, ya que el candidato de AMVP solo contiene un vector de movimiento. En la modalidad de AMVP, los vectores de movimiento predichos se pueden refinar aún más.

[0030] Un candidato de fusión corresponde a un conjunto completo de información de movimiento, mientras que un candidato de AMVP puede contener solo un vector de movimiento para una dirección de predicción específica e índice de referencia. Los candidatos para ambas modalidades se obtienen de manera similar a partir de los mismos bloques vecinos espaciales y temporales.

[0031] Como se describe a continuación con respecto a la figura 5, los candidatos de MV espacial se obtienen de los bloques vecinos que se muestran en la figura 5, para una PU específica (PU0). Sin embargo, los procedimientos que generan los candidatos de los bloques difieren para las modalidades de fusión y de AMVP.

5 **[0032]** En la modalidad de fusión, se pueden obtener hasta cuatro candidatos espaciales de MV con los órdenes mostrados en la figura 5 (a) con números. El orden es el siguiente: izquierda (0), arriba (1), arriba derecha (2), abajo izquierda (3) y arriba izquierda (4), como se muestra en la figura 5 (a).

10 **[0033]** En la modalidad de AMVP, los bloques vecinos se dividen en dos grupos: un grupo izquierdo que incluye los bloques 0 y 1, y un grupo superior que incluye los bloques 2, 3 y 4, como se muestra en la figura 5 (b). Para cada grupo, el candidato potencial en un bloque vecino, que se refiere a la misma imagen de referencia que la indicada por el índice de referencia señalado, tiene la prioridad más alta para ser elegido para formar un candidato final del grupo. Es posible que todos los bloques vecinos no contengan un vector de movimiento que apunte a la misma imagen de referencia. Por lo tanto, si no se puede encontrar dicho candidato, el primer candidato disponible se ajustará a escala para formar el candidato final, por lo que las diferencias de distancia temporal se pueden compensar.

15 **[0034]** Como se describe a continuación con respecto a la figura 6, la Copia intra-Bloque (BC) se ha incluido en SCC. Un ejemplo de Intra-BC se muestra en la figura 6, en donde la CU actual se predice a partir de un bloque ya decodificado de la imagen / fragmento actual. El tamaño actual del bloque de Intra-BC puede ser tan grande como un tamaño de CU, que varía de 8x8 a 64x64, aunque, en algunas aplicaciones, pueden aplicarse restricciones adicionales.

20 **[0035]** En algunos ejemplos, se puede usar un proceso de transformación de color-espacio en el bucle para señales residuales, para secuencias en formato 4: 4: 4 de croma. Este procedimiento transforma las señales de error de predicción, en formato de croma RGB / YUV, en las de un color-espacio subóptimo. Con esta etapa adicional, la correlación entre los componentes de color podría reducirse aún más. La matriz de transformación se obtiene de los valores de muestra de píxeles para cada unidad de codificación mediante una descomposición en valores singulares (SVD). La transformación de color-espacio se aplica al error de predicción de ambas modalidades intra e inter.

25 **[0036]** En algunos ejemplos, cuando el proceso de transformación de color-espacio se aplica a la modalidad inter, el residuo se convierte primero a un dominio diferente con la matriz de transformación obtenida. Después de la conversión de color-espacio, las etapas de codificación convencional, tales como DCT / DST, cuantización y codificación por entropía se realizan en orden.

30 **[0037]** En algunos ejemplos, cuando el proceso de transformación de color-espacio se aplica a la modalidad intra, la predicción y el bloque actual, respectivamente, se convierten primero a un dominio diferente con la matriz de transformación obtenida. Después de la conversión de color-espacio, el residuo entre el bloque actual y su predictor se transforma adicionalmente con DCT / DST, se cuantiza y se codifica por entropía.

35 **[0038]** En el funcionamiento directo, una matriz de transformación de color-espacio se aplica a los tres planos G, B y R de la siguiente manera:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} G \\ B \\ R \end{bmatrix} = \begin{bmatrix} P \\ Q \\ S \end{bmatrix}.$$

40 **[0039]** Los valores resultantes se recortan dentro del rango de la especificación de la HEVC porque, en algunos ejemplos, los valores se amplían hasta $\sqrt{3}$ veces. En el funcionamiento inverso, una matriz de transformación de color-espacio se aplica a los tres componentes P', Q' y R' de la siguiente manera:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}^t \begin{bmatrix} P' \\ Q' \\ S' \end{bmatrix} = \begin{bmatrix} G' \\ B' \\ R' \end{bmatrix}.$$

45 **[0040]** Como se describe a continuación con respecto a la figura 7, se puede obtener una matriz de transformación a partir de los valores de muestra de referencia. Se pueden utilizar diferentes muestras de referencia para el caso intra y el caso inter. Para el caso de un bloque intracodificado, se muestran un bloque de destino y muestras de referencia en la figura 7. En este ejemplo, el bloque de destino consiste en muestras cuadrículas de tamaño 8x8 y las muestras de referencia son muestras de puntos y rayas.

50 **[0041]** Para el caso de un bloque intercodificado, las muestras de referencia para la obtención de la matriz son las mismas que las muestras de referencia para la compensación de movimiento. Para realizar la operación de desplazamiento, se submuestran las muestras de referencia en el bloque AMP de modo que el número de muestras

se convierta en una potencia de dos. Por ejemplo, el número de muestras de referencia en un bloque de tamaño 12x16 se reduce en 2/3.

5 **[0042]** De acuerdo con las técnicas de la presente divulgación, el proceso de transformación de color-espacio puede aplicarse para cada CU. Por lo tanto, no es necesario señalar si el proceso de transformación se invoca o no. Además, tanto el lado del codificador como el del decodificador obtienen la matriz de transformación con el mismo procedimiento para evitar la sobrecarga para señalar la matriz de transformación.

10 **[0043]** De acuerdo con las técnicas de la presente divulgación, se utilizan procesos de transformación de color-espacio, tales como matrices de transformación de color-espacio. Una de estas matrices es la matriz de transformación YCbCr, que es:

Directo:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0,2126 & 0,7152 & 0,0722 \\ -0,1172 & -0,3942 & 0,5114 \\ 0,5114 & -0,4645 & -0,0469 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

15

Inverso:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1,5397 \\ 1 & -0,1831 & -0,4577 \\ 1 & 1,8142 & 0 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix}$$

20

[0044] Otra matriz de este tipo es la matriz de transformación YCoCg, que es:

Directo:

$$\begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1/2 & 0 & -1/2 \\ -1/4 & 1/2 & -1/4 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Inverso:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 \\ 1 & 0 & 1 \\ 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix}$$

25

[0045] Otra matriz de este tipo es la matriz YCoCg-R, que es una versión revisada de la matriz YCoCg-R que ajusta a escala los componentes de Co y Cg en un factor de dos. Mediante el uso de una técnica de elevación, la transformación directa e inversa podría lograrse mediante las siguientes ecuaciones:

30

Directo:

$$\begin{aligned} Co &= R - B \\ t &= B + \lfloor Co/2 \rfloor \\ Cg &= G - t \\ Y &= t + \lfloor Cg/2 \rfloor \end{aligned}$$

Inverso:

$$\begin{aligned} t &= Y - \lfloor Cg/2 \rfloor \\ G &= Cg + t \\ B &= t - \lfloor Co/2 \rfloor \\ R &= B + Co \end{aligned}$$

35

[0046] En las ecuaciones y matrices anteriores, las transformaciones directas pueden realizarse durante el proceso de codificación (por ejemplo, mediante un codificador de vídeo), y las transformaciones inversas pueden realizarse en el proceso de decodificación (por ejemplo, mediante un decodificador de vídeo).

40

[0047] En la codificación de vídeo tradicional, se asume que las imágenes tienen un tono continuo y son espacialmente llanas. Sobre la base de estas hipótesis, se han desarrollado varias herramientas (por ejemplo, transformación basada en bloques, filtrado, etc.) que han demostrado un buen rendimiento para vídeos con contenido natural. Sin embargo, en ciertas aplicaciones (por ejemplo, un escritorio remoto, pantallas de trabajo colaborativo y pantallas inalámbricas), el contenido de la pantalla generado por ordenador puede ser el contenido dominante a comprimir. Este tipo de contenido tiende a tener un tono discreto y presenta líneas nítidas y fronteras de objetos de alto contraste. Es posible que la hipótesis de tono continuo y llaneza ya no se aplique y, por lo tanto, las técnicas de

45

codificación de vídeo tradicionales no funcionen de manera eficaz.

[0048] La codificación en modalidad de paleta se puede utilizar para superar las deficiencias anteriores. Ejemplos de las técnicas de codificación de paleta se describen en la Solicitud Provisional Estadounidense con Número de serie 61 / 810,649, presentada el 10 de abril de 2013. Para cada CU, se puede obtener una paleta, que incluye los valores de píxel más dominantes en la CU actual. El tamaño y los elementos de la paleta se transmiten primero. Después de la transmisión, los píxeles en la CU se codifican de acuerdo con un cierto orden de recorrido. Para cada ubicación, un elemento sintáctico, tal como un indicador, indicador_paleta, se transmite primero para indicar si el valor de píxel está en la paleta ("modalidad de racha") o no ("modalidad de píxel").

[0049] En "modalidad de racha", el índice de paleta puede ser señalado, seguido por la "racha". La racha es un elemento sintáctico que indica el número de píxeles consecutivos en un orden de recorrido que tienen el mismo valor de índice de paleta que el píxel que se está codificando actualmente. Si varios píxeles en sucesión inmediata en el orden de recorrido tienen el mismo valor de índice de paleta, entonces el elemento sintáctico puede indicar "modo de racha", tal como el indicador_paleta. Se puede determinar un valor de contador, que es igual al número de píxeles que siguen al píxel actual que tienen el mismo valor de índice de paleta que el píxel actual, y la racha se fija igual al valor del contador. Ni el indicador_paleta ni el índice de paleta deben transmitirse para las siguientes posiciones que están cubiertas por la "racha", ya que todas tienen el mismo valor de píxel. En el lado del decodificador, solo se decodificaría el primer valor de índice de paleta para el píxel actual, y el resultado se duplicaría para cada píxel en la "racha" de los píxeles indicados en el elemento sintáctico "racha".

[0050] En "modalidad de píxel", el valor de muestra de píxel se transmite para esta posición. Si el elemento sintáctico, tal como indicador_paleta, indica "modalidad de píxel", entonces el valor del índice de paleta solo se determina para el píxel que se está decodificando.

[0051] Las técnicas convencionales pueden adolecer de diversos problemas. Por ejemplo, se puede invocar una transformación de color-espacio que no tenga en cuenta las características de secuencia y la diversidad local. Por lo tanto, el rendimiento de codificación puede ser subóptimo. En otro ejemplo, la obtención de una matriz de transformación puede ser necesaria en el decodificador, lo que aumenta significativamente la complejidad del decodificador. Además, la matriz de transformación puede obtenerse utilizando los píxeles reconstruidos espaciales o el predictor de las PU intercodificadas. Sin embargo, la eficacia de la matriz de transformación puede reducirse cuando el tamaño de PU es relativamente pequeño, la predicción no es muy precisa o los píxeles vecinos no están disponibles. Las técnicas de esta divulgación pueden superar uno o más de estos problemas.

[0052] Un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, puede ejecutar las técnicas descritas en esta divulgación. En general, esta divulgación describe técnicas para codificar bloques de vídeo usando un proceso de conversión de color-espacio. Un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, puede determinar una modalidad de codificación usada para codificar los datos de vídeo. La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. El codificador de vídeo puede determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. El codificador de vídeo puede aplicar el proceso de transformación de color-espacio en la codificación de los datos de vídeo. Al decodificar los datos de vídeo, independientemente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas, el codificador de vídeo puede aplicar el mismo proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación.

[0053] Esta divulgación describe técnicas que pueden mejorar el rendimiento de codificación de la transformación de color-espacio en bucle y pueden reducir la complejidad del decodificador. La figura 1 es un diagrama de bloques que ilustra un sistema a modo de ejemplo de codificación y decodificación de vídeo 10 que puede utilizar técnicas para la codificación de contenido de pantalla, en donde se utiliza un formato de muestreo de croma alto. Como se muestra en la figura 1, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados, para ser decodificados en un momento posterior por un dispositivo de destino 14. En particular, el dispositivo de origen 12 proporciona los datos de vídeo al dispositivo de destino 14 mediante un medio legible por ordenador 16. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos manuales de teléfono tales como los denominados teléfonos «inteligentes», los denominados paneles «inteligentes», televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de transmisión en continuo de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0054] El dispositivo de destino 14 puede recibir los datos de vídeo codificados, a decodificar mediante el medio legible por ordenador 16. El medio legible por ordenador 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el medio legible por ordenador 16 puede comprender un medio de comunicación que habilita el dispositivo de origen 12 para transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de

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

[0055] En algunos ejemplos, pueden emitirse datos codificados desde la interfaz de salida 22 hasta un dispositivo de almacenamiento. De forma similar, se puede acceder a los datos codificados procedentes del dispositivo de almacenamiento mediante una interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera entre una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco duro, discos Blu-ray, discos DVD, discos 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 almacenamiento puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda almacenar los datos de vídeo codificados generados por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados procedentes del dispositivo de almacenamiento, mediante transmisión continua o descarga. 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 dispositivo de destino 14. Entre los ejemplos de servidores de ficheros se incluyen un servidor de la Red (por ejemplo, para una sede de la Red), un servidor del FTP, dispositivos de almacenamiento conectados en red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión cableada (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0056] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo como soporte de cualquiera entre una diversidad de aplicaciones de multimedios, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo por Internet, tales como el flujo de transmisión adaptativo dinámico sobre HTTP (DASH), el vídeo digital que se codifica en un medio de almacenamiento de datos, la decodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para prestar soporte a la transmisión de vídeo unidireccional o bidireccional para prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la difusión de vídeo y/o la videotelefonía.

[0057] En el ejemplo de la figura 1, el dispositivo de origen 12 incluye un origen de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. El dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. De acuerdo con esta divulgación, el codificador de vídeo 20 del dispositivo de origen 12 puede configurarse para aplicar las técnicas para codificar bloques de vídeo usando un proceso de conversión de color-espacio. En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde un origen de vídeo externo 18, tal como una cámara externa. Asimismo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0058] El sistema ilustrado 10 de la figura 1 es simplemente un ejemplo. Las técnicas para codificar bloques de vídeo usando un proceso de conversión de color-espacio pueden ser realizadas por cualquier dispositivo de codificación y / o decodificación de vídeo digital. Aunque, en general, las técnicas de esta divulgación se llevan a cabo mediante un dispositivo de codificación de vídeo, las técnicas también pueden llevarse a cabo mediante un codificador/decodificador de vídeo, denominado habitualmente "CÓDEC". Además, las técnicas de esta divulgación también pueden llevarse a cabo mediante un preprocesador de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 son simplemente ejemplos de dichos dispositivos de codificación en los que el dispositivo de origen 12 genera datos de vídeo codificados para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de manera esencialmente simétrica, de modo que cada uno de los dispositivos 12, 14 incluya componentes de codificación y de decodificación de vídeo. Por tanto, el sistema 10 puede prestar soporte a la transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo, para el flujo de vídeo, la reproducción de vídeo, la difusión de vídeo o la videotelefonía.

[0059] El origen de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado previamente y/o una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenidos de vídeo. Como alternativa adicional, el origen de vídeo 18 puede generar datos, basados en gráficos por ordenador, como el vídeo de origen, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, 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, como se ha mencionado anteriormente, las técnicas descritas en esta

divulgación pueden ser aplicables a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el vídeo capturado, precapturado o generado por ordenador puede codificarse mediante el codificador de vídeo 20. La información de vídeo codificada puede entonces emitirse mediante la interfaz de salida 22 hacia un medio legible por ordenador 16.

[0060] El medio legible por ordenador 16 puede incluir medios transitorios, tales como una difusión inalámbrica o una transmisión por red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorio), tales como un disco duro, una unidad de memoria flash, un disco compacto, un disco de vídeo digital, un disco Blu-ray u otros medios legibles por ordenador. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y proporcionar los datos de vídeo codificados al dispositivo de destino 14, por ejemplo, mediante transmisión por red. De manera similar, un dispositivo informático de una utilidad de producción de un medio, tal como una utilidad de grabación de discos, puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y producir un disco que contiene los datos de vídeo codificados. Por lo tanto, puede entenderse que el medio legible por ordenador 16 incluye uno o más medios legibles por ordenador de diversas formas, en varios ejemplos.

[0061] La interfaz de entrada 28 del dispositivo de destino 14 recibe información desde el medio legible por ordenador 16. La información del medio legible por ordenador 16 puede incluir información sintáctica definida por el codificador de vídeo 20, que también usa el decodificador de vídeo 30, que incluye elementos sintácticos que describen características y/o procesamiento de bloques y otras unidades codificadas, por ejemplo, grupos de imágenes (GOP). El dispositivo de visualización 32 muestra los datos de vídeo decodificados a un usuario, y puede comprender cualquiera entre diversos dispositivos de visualización, tales como un tubo de rayos catódicos (CRT), 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.

[0062] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con una norma de codificación de vídeo, tal como la norma de codificación de vídeo de alta eficacia (HEVC), actualmente en fase de desarrollo, y pueden ajustarse al modelo de prueba de la HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con otras normas de propiedad o industriales, tales como la norma ITU-T H.264, alternativamente denominada MPEG-4, parte 10, Codificación Avanzada de Vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de codificación de vídeo incluyen MPEG-2 e ITU-T H.263. Aunque no se muestra en la figura 1, en algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar integrados, cada uno de ellos, con un codificador y un decodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos separados. Si procede, las unidades MUX-DEMUX pueden ajustarse al protocolo de multiplexado ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0063] La norma ITU-T H.264/MPEG-4 (AVC) fue formulada por el Grupo de Expertos en Codificación de Vídeo (VCEG) de la ITU-T junto con el Grupo de Expertos en Imágenes en Movimiento (MPEG) de la ISO/IEC, como el producto de una asociación colectiva conocida como Equipo Mixto de Vídeo (JVT, por sus siglas en inglés). En algunos aspectos, las técnicas descritas en esta divulgación pueden aplicarse a dispositivos que se ajustan en general a la norma H.264. La norma H.264 se describe en la Recomendación de la ITU-T H.264, Codificación Avanzada de vídeo para servicios audiovisuales genéricos, por el Grupo de Estudio de la ITU-T, y con fecha de marzo de 2005, que puede denominarse en el presente documento norma H.264 o memoria descriptiva H.264, o norma o memoria descriptiva H.264/AVC. El Equipo Mixto de vídeo (JVT) continúa trabajando en ampliaciones de la norma H.264/MPEG-4 AVC.

[0064] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse, cada uno de ellos, como cualquiera entre una variedad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de estos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede estar incluido en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) combinado en un dispositivo respectivo.

[0065] El equipo JCT-VC está trabajando en la elaboración de la norma HEVC. La iniciativa de normalización de la HEVC se basa en un modelo en evolución de un codificador de vídeo denominado modelo de prueba de la HEVC (HM). El HM supone varias capacidades adicionales de los codificadores de vídeo respecto a dispositivos existentes, de acuerdo, por ejemplo, a la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modalidades de codificación de intrapredicción, el HM puede proporcionar hasta treinta y tres modalidades de codificación de intrapredicción.

[0066] En general, el modelo de explotación del HM describe que una trama o imagen de vídeo puede dividirse en

una secuencia de bloques de árbol o unidades de codificación de máximo tamaño (LCU), que incluyen muestras tanto de luma como de croma. Los datos sintácticos dentro de un flujo de bits pueden definir un tamaño para la LCU, que es la máxima unidad de codificación en lo que respecta al número de píxeles. Un fragmento incluye un número de bloques de árbol consecutivos en orden de codificación. Una trama o imagen de vídeo puede dividirse en uno o más fragmentos. Cada bloque arbolado puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuádruple. En general, una estructura de datos de árbol cuádruple incluye un nodo por CU, con un nodo raíz correspondiente al bloque arbolado. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hoja, cada uno de los cuales corresponde a una de las sub-CU.

[0067] Cada nodo de la estructura de datos en árbol cuádruple puede proporcionar datos sintácticos para la CU correspondiente. Por ejemplo, un nodo en el árbol cuádruple puede incluir un indicador de división, que indica si la CU correspondiente al nodo está dividida en varias sub-CU. Los elementos sintácticos para una CU pueden definirse de manera recursiva y pueden depender de si la CU está dividida en varias sub-CU. Si una CU no está dividida adicionalmente, se denomina CU hoja. En esta divulgación, cuatro sub-CU de una CU hoja también se denominarán CU hojas incluso aunque no haya ninguna división explícita de la CU hoja original. Por ejemplo, si una CU con un tamaño de 16x16 no se divide adicionalmente, las cuatro sub-CU de tamaño 8x8 también se denominarán CU hojas aunque la CU de tamaño 16x16 no se haya dividido nunca.

[0068] Una CU tiene un propósito similar a un macrobloque de la norma H.264, excepto porque una CU no tiene una distinción de tamaño. Por ejemplo, un bloque arbolado puede dividirse en cuatro nodos secundarios (también denominados sub-CU) y cada nodo secundario puede a su vez ser un nodo principal y dividirse en otros cuatro nodos secundarios. Un nodo secundario final, no dividido, denominado nodo hoja del árbol cuádruple, comprende un nodo de codificación, también denominado CU hoja. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces en que puede dividirse un bloque arbolado, denominado profundidad de CU máxima, y también pueden definir un tamaño mínimo de los nodos de codificación. Por consiguiente, un flujo de bits también puede definir una unidad de codificación de tamaño mínimo (SCU). Esta divulgación utiliza el término "bloque" para referirse a cualquiera entre una CU, PU o TU, en el contexto de la HEVC, o a estructuras de datos similares en el contexto de otras normas (por ejemplo, macrobloques y subbloques de los mismos en la norma H.264/AVC).

[0069] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformación (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe ser de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque arbolado, con un máximo de 64x64 píxeles o más. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Las modalidades de división pueden diferir entre si la CU está codificada en modalidad de salto o directa, codificada en modalidad de intrapredicción o codificada en modalidad de interpredicción. Las PU pueden dividirse para tener forma no cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuádruple. Una TU puede tener forma cuadrada o no cuadrada (por ejemplo, rectangular).

[0070] La norma HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU habitualmente se basa en el tamaño de las PU dentro de una CU dada, definida para una LCU dividida, aunque puede que no sea siempre así. Las TU son habitualmente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas mediante una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformación (TU). Los valores de diferencia de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformación, que pueden cuantizar.

[0071] Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa una zona espacial correspondiente a la totalidad, o a una parte, de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Además, una PU incluye datos relacionados con la predicción. Por ejemplo, cuando la PU está codificada en intramodalidad, pueden incluirse datos para la PU en un árbol cuádruple residual (RQT), que pueden incluir datos que describen una modalidad de intrapredicción para una TU correspondiente a la PU. Como otro ejemplo, cuando la PU está codificada en intermodalidad, la PU puede incluir datos que definen uno o más vectores de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, lista 0, lista 1 o lista C) para el vector de movimiento.

[0072] Una CU hoja que tiene una o más PU también puede incluir una o más unidades de transformación (TU). Las unidades de transformación pueden especificarse usando un RQT (también denominado estructura de árbol cuádruple de TU), como se ha expuesto anteriormente. Por ejemplo, un indicador de división puede indicar si una CU hoja está dividida en cuatro unidades de transformación. A continuación, cada unidad de transformación puede dividirse adicionalmente en sub-TU adicionales. Cuando una TU no se divide adicionalmente, puede denominarse una TU hoja. En general, en lo que respecta a la intracodificación, todas las TU hojas que pertenecen a una CU hoja comparten la

misma modalidad de intrapredicción. Es decir, la misma modalidad de intrapredicción se aplica en general para calcular valores predichos para todas las TU de una CU hoja. En lo que respecta a la intracodificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando la modalidad de intrapredicción, como una diferencia entre la parte de la CU correspondiente a la TU y el bloque original. Una TU no está necesariamente limitada al tamaño de una PU. De este modo, las TU pueden ser mayores o menores que una PU. En lo que respecta a la intracodificación, una PU puede estar cosituada con una TU hoja correspondiente para la misma CU. En algunos ejemplos, el tamaño máximo de una TU hoja puede corresponder al tamaño de la CU hoja correspondiente.

[0073] Además, las TU de las CU hojas también pueden asociarse a respectivas estructuras de datos de árbol cuádruple, denominadas árboles cuádruples residuales (RQT). Es decir, una CU hoja puede incluir un árbol cuádruple que indica cómo la CU hoja está dividida en las TU. El nodo raíz de un árbol cuádruple de TU corresponde en general a una CU hoja, mientras que el nodo raíz de un árbol cuádruple de CU corresponde en general a un bloque arbolado (o LCU). Las TU del RQT que no están divididas se denominan TU hojas. En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a no ser que se indique lo contrario.

[0074] Una secuencia de vídeo incluye habitualmente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, en una cabecera de una o más de las imágenes, o en otras ubicaciones, que describen una serie 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. Un codificador de vídeo 20 actúa habitualmente sobre bloques de vídeo dentro de fragmentos de vídeo individuales con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden tener tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

[0075] En un ejemplo, el HM da soporte a la predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es $2N \times 2N$, el HM da soporte a la intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$, y a la interpredicción en tamaños de PU simétricas de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también da soporte a la división asimétrica para la interpredicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida en el 25% y el 75%. La parte de la CU correspondiente a la división del 25% está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así pues, por ejemplo, " $2N \times nU$ " se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente con una PU de tamaño $2N \times 0,5N$ por encima y una PU de tamaño $2N \times 1,5N$ por debajo.

[0076] En esta divulgación, " $N \times N$ " y " N por N " pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de tamaño 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ tiene, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles en un bloque pueden disponerse en filas y columnas. Además, no es necesario que los bloques tengan necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

[0077] Tras la codificación intrapredictiva o interpredictiva utilizando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos sintácticos que describen un procedimiento o modalidad de generación de datos de píxeles predictivos en el dominio espacial (también denominado el dominio de píxeles) y las TU pueden comprender coeficientes en el dominio de transformación, tras la aplicación de una transformación, por ejemplo, una transformación de coseno discreta (DCT), una transformación entera, una transformación de ondículas o una transformación conceptualmente similar, a los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxel entre píxeles de la imagen no codificada y los valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU incluyendo los datos residuales para la CU y, a continuación, transformar las TU para generar coeficientes de transformación para la CU.

[0078] Tras cualquier transformación para generar coeficientes de transformación, el codificador de vídeo 20 puede realizar la cuantización de los coeficientes de transformación. La cuantización se refiere, en general, 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 compresión adicional. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes. Por ejemplo, un valor de n bits puede redondearse hacia abajo hasta un valor de m bits durante la cuantización, donde n es mayor que m .

[0079] Después de la cuantización, el codificador de vídeo puede recorrer los coeficientes de transformación, produciendo un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformación cuantizados. El recorrido puede estar diseñado para colocar los coeficientes de energía más alta (y por lo tanto de menor frecuencia) en la parte frontal de la matriz y para colocar los coeficientes de energía más baja (y por lo tanto de mayor frecuencia) en la parte posterior de la matriz. En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de recorrido predefinido para recorrer los coeficientes de transformación cuantizados a fin de generar un vector

en serie que se pueda codificar por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar un recorrido adaptativo. Después de recorrer los coeficientes de transformación cuantizados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo a la codificación de longitud variable adaptable al contexto (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación aritmética binaria adaptable al contexto basada en la sintaxis (SBAC), la codificación por entropía por división de intervalos de probabilidad (PIPE) o con otra metodología de codificación por entropía. El codificador de vídeo 20 también puede realizar la codificación por entropía de los 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.

[0080] Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto de un modelo de contexto a un símbolo que se va a transmitir. El contexto puede indicar, por ejemplo, si los valores contiguos del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo que se va a transmitir. Las palabras de código en la VLC pueden construirse de tal manera que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.

[0081] De acuerdo con las técnicas de esta divulgación, un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, puede determinar una modalidad de codificación usada para codificar los datos de vídeo. La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. El codificador de vídeo puede determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. El codificador de vídeo puede aplicar el proceso de transformación de color-espacio en la codificación de los datos de vídeo. Al decodificar los datos de vídeo, independientemente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas, el codificador de vídeo puede aplicar el mismo proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación.

[0082] Un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, puede realizar cualquiera de las técnicas según lo descrito en relación con las figuras 1 a 9. Al utilizar la misma matriz de transformación inversa del color-espacio, un decodificador de vídeo puede ser capaz de decodificar de manera más eficaz los datos de vídeo cuando se utiliza la transformación del color-espacio. En diferentes casos, puede ser más eficaz que un codificador de vídeo use una modalidad de codificación con pérdidas. En otros casos, puede ser más eficaz que un decodificador de vídeo use una modalidad de codificación sin pérdidas. Para una calidad de imagen superior, se puede implementar una modalidad de codificación sin pérdidas en el lado de decodificación, y se puede usar una matriz de codificación sin pérdidas en el lado del decodificador, independientemente de si los datos de vídeo están codificados con una modalidad de codificación con pérdidas o una modalidad de codificación sin pérdidas. Como tal, puede aumentar la eficacia de un decodificador de vídeo implementar la misma matriz de transformación inversa sin pérdidas de color-espacio, sin importar si los datos de vídeo están codificados con una modalidad de codificación con pérdidas o una modalidad de codificación sin pérdidas, eliminando la etapa de determinar si los datos de vídeo se codifican con una modalidad de codificación con pérdidas o una modalidad de codificación sin pérdidas, aumentando de ese modo la eficacia de codificación global del sistema y reduciendo el consumo de energía.

[0083] Se puede utilizar una diversidad de elementos sintácticos en alineación con las técnicas de la divulgación actual. Estos elementos sintácticos pueden incluir:

Tabla 1

	Descriptor
conjunto_parámetros_sec_rbsp() {	
identificador_conjunto_parámetros_vídeo_sps	u(4)
máx_sub_capas_menos1_sps	u(3)
indicador_anidamiento_identificador_temporal_sps	u(1)
nivel_categoría_perfil (máx_sub_capas_menos1_sps)	
...	
indicador_parámetros_vui_presentes	u(1)
si(indicador_parámetros_vui_presentes)	
parámetros_vui()	
indicador_extensión_presente_sps	u(1)
si (indicador_extensión_presente_sps) {	

conjunto_parámetros_sec_rbsp() {	Descriptor
para(i = 0; i < 1; i++)	
indicador_extensión_sps [i]	u(1)
7bits_extensión_sps	u(7)
si (indicador_extensión_sps [0]) {	
indicador_rotación_habilitada_salto_transformación	u(1)
indicador_rotación_habilitada_salto_transformación	u(1)
indicador_copia_intra_bloque_habilitada	u(1)
indicador_rdpcm_implicito_habilitado	u(1)
indicador_rdpcm_explicito_habilitado	u(1)
indicador_procesamiento_precisión_extendida	u(1)
indicador_intra_allanamiento_inhabilitado	u(1)
indicador_desplazamientos_alta_precisión_habilitados	u(1)
indicador_adaptación_rápida_elevación_habilitado	u(1)
indicador_alineación_omisión_cabac_habilitada	u(1)
indicador_transformación_color_habilitada (nuevo)	u(1)
}	
si (extensión_7bits_sps)	
mientras(más_datos_rbsp())	
indicador_datos_extensión_sps	u(1)
}	
bits_de_cola_rbsp()	
}	
unidad_codificación(x0, y0, log2TamañoCb){	Descriptor
si (indicador_omisión_transcuant_habilitada)	
indicador_omisión_transcuant_cu	ae(v)
si(tipo_fragmento != I)	
indicador_omisión_cu [x0] [y0]	ae(v)
nCbS = (1 <<log2TamañoCb)	
si (indicador_omisión_cu [x0] [y0])	
unidad_predicción(x0, y0, nCbS, nCbS)	
en caso contrario {	
si (indicador_intra_copia_bloque_habilitada)	
indicador_intra_bc [x0] [y0]	ae(v)
si (tipo_fragmento! = I &&! indicador_intra_bc [x0] [y0])	
indicador_modalidad_pred	ae(v)
si (ModalidadPredCu [x0] [y0]! = INTRA_MODALIDAD indicador_intra_bc [x0] [y0] log2TamañoCb == MinCbLog2TamañoY)	
modalidad_part	ae(v)
unidad_codificación(x0, y0, log2TamañoCb){	Descriptor

ES 2 702 950 T3

	Descriptor
conjunto_parámetros_sec_rbsp() {	
si (ModalidadPredCu [x0] [y0] == INTRA_MODALIDAD) {	
si(ModalidadPart == PART_2Nx2N && indicador_pcm_habilitado && ! indicador_intra_bc [x0] [y0] && log2TamañoCb >= Log2MinIpcmTamañoCbY && log2TamañoCb <= Log2MaxIpcmTamañoCbY)	
indicador_pcm [x0] [y0]	ae(v)
si (indicador_pcm [x0] [y0]) {	
mientras(!alineado_a_octeto())	
bit_cero_alineación_pcm	f(1)
muestra_pcm(x0, y0, log2TamañoCb)	
} en caso contrario si (indicador_intra_bc [x0] [y0]) {	
codificación_mvd (x0, y0, 2)	
si(ModalidadPart == PART_2NxN)	
codificación_mvd (x0, y0 + (nCbs / 2), 2)	
en caso contrario si (ModalidadPart == PART_Nx2N)	
codificación_mvd (x0 + (nCbs / 2), y0, 2)	
en caso contrario si (ModalidadPart == PART_NxN) {	
codificación_mvd (x0 + (nCbs / 2), y0, 2)	
codificación_mvd (x0, y0 + (nCbs / 2), 2)	
codificación_mvd (x0 + (nCbs / 2), y0 + (nCbs / 2), 2)	
}	
} en caso contrario {	
desplazamientoPb = (ModalidadPart == PART_NxN) ? (nCbs / 2) : nCbS	
para (j = 0; j <nCbS; j = j + desplazamientoPb)	
para (i = 0; i <nCbS; i = i + desplazamientoPb)	
indicador_pred_prev_intra_luma [x0 + i][y0 + j]	ae(v)
para (j = 0; j <nCbS; j = j + desplazamientoPb)	
para (i = 0; i <nCbS; i = i + desplazamientoPb)	
si (indicador_pred_prev_intra_luma [x0 + i] [y0 + j])	
idx_mpm [x0 + i] [y0 + j]	ae(v)
en caso contrario	
modalidad_pred_intra_luma_rem [x0 + i][y0 + j]	ae(v)
si (TipoFormaciónCroma == 3)	
para (j = 0; j <nCbS; j = j + desplazamientoPb)	
para (i = 0; i <nCbS; i = i + desplazamientoPb)	
modalidad_pred_intra_croma [x0 + i] [y0 + j]	ae(v)
en caso contrario si (TipoFormaciónCroma != 0)	
modalidad_pred_intra_croma [x0][y0]	ae(v)
unidad_codificación(x0, y0, log2TamañoCb){	Descriptor
}	

en el flujo de bits para una CU, se puede omitir el proceso de transformación de color-espacio.

[0088] Además, un indicador en cada cabecera de conjunto de parámetros de secuencia (SPS) / conjunto de parámetros de imagen (PPS) / fragmento se puede señalar para controlar el uso de la transformación de color-espacio. Cuando el indicador en la cabecera de SPS / PPS / fragmento es igual a 0, la señalización del indicador de nivel de CU se puede omitir, respectivamente, en todas las CU en la correspondiente secuencia / imagen / fragmento. Alternativamente, la transformación de color-espacio se puede realizar en el nivel de PU o en el nivel de TU, en donde cada PU / TU tiene un indicador para indicar el uso de la transformación de color-espacio.

[0089] En un ejemplo, se puede aplicar una transformación YCoCg modificada en el proceso de codificación y decodificación. La transformación YCoCg modificada puede definirse de la siguiente manera:

Directa:

$$\begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 0 & -2 \\ -1 & 2 & -1 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix},$$

Inversa: temp = Y-Cg;

$$G = (Y + Cg + desplazamiento) \gg 2;$$

$$B = (temp - Co + desplazamiento) \gg 2;$$

R = (temp + Co + desplazamiento) >> 2; en donde el desplazamiento es igual a 2.

[0090] Alternativamente, la transformación YCoCg modificada se puede definir de la siguiente manera:

Directa:

$$\begin{bmatrix} Y \\ Co \\ Cg \end{bmatrix} = \begin{bmatrix} 1/2 & 1 & 1/2 \\ 1 & 0 & -1 \\ -1/2 & 1 & -1/2 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Inversa: temp = Y-Cg;

$$G = (Y + Cg + desplazamiento) \gg 1;$$

$$B = (temp - Co + desplazamiento) \gg 1;$$

$$R = (temp + Co + desplazamiento) \gg 1; \text{ en donde el desplazamiento es igual a } 1 \text{ o } 0.$$

[0091] Se puede deducir que el parámetro de cuantización correspondiente utilizado por las CU o los bloques que están codificados con la transformación de color es igual a (dQP + 2), mientras que se puede deducir que el parámetro utilizado por las CU o los bloques sin transformación de color es igual a dQP. Simultáneamente, la profundidad de bits se puede aumentar adicionalmente en 1 en ambos procesos de cuantización y transformación.

[0092] Alternativamente, se aplican dos etapas de transformación directa / inversa con la transformación YCoCg original sin cambios más un proceso de normalización. Para la transformación directa, se aplica primero la transformación directa YCoCg original. Luego, para cada componente i, es decir, Y, Co y Cg, el componente se reinicia en (i * factor_directo + desplazamiento) >> PRECISIÓN_BIT, en donde PRECISIÓN_BIT es un entero sin signo y factor_directo depende de PRECISIÓN_BIT. En un ejemplo, factor_directo es igual a 1 / √6 * (1 < <BITS_TRANS) + 0,5, PRECISIÓN_BIT es igual a 15, desplazamiento es igual a (1 << (PRECISIÓN_BIT - 1)).

[0093] Para la transformación inversa, se aplica primero la transformación inversa YCoCg original. Luego, para cada componente i, es decir, Y, Co y Cg, el componente se reinicia en (i * factor_inverso + desplazamiento) >> PRECISIÓN_BIT, en donde PRECISIÓN_BIT es la misma que la usada en la transformación directa, y factor_inverso depende de PRECISIÓN_BIT. En un ejemplo, factor_inverso es igual a (√6 / 4 * (1 < <BITS_TRANS) + 0,5), desplazamiento es igual a (1 << (PRECISIÓN_BIT - 1)).

[0094] Alternativamente / además, el indicador de nivel de CU puede señalizarse solo cuando al menos uno de los indicadores de bloque codificados en tres componentes de color (es decir, cbf_luma, cbf_cb y cbf_cr) es igual a 1. Alternativamente, el residuo modificado después de la transformación de color-espacio puede modificarse más para asegurarse de que el rango del residuo en las CU o los bloques con transformación de color-espacio sea el mismo

que el del residuo en las CU / los bloques sin la transformación de color-espacio. En un ejemplo, se aplica una operación de recorte.

5 **[0095]** La matriz de transformación de color-espacio puede ser independiente de los píxeles reconstruidos. En cambio, puede depender de la modalidad de codificación con pérdidas o sin pérdidas. En un ejemplo, cuando la CU se codifica con la modalidad con pérdidas, YCoCg puede aplicarse mientras que YCoCg-R se aplica cuando la CU se codifica con la modalidad sin pérdidas. Además, cuando se aplica YCoCg-R, la profundidad de bits de los componentes Co y Cg puede aumentar en 1. En otro ejemplo, la matriz de transformación puede depender de las modalidades intra, inter / Intra BC. En este caso, se puede especificar una matriz predefinida para cada modalidad, tanto en el codificador como en el decodificador, y la misma matriz se puede usar tanto en el codificador como en el decodificador, de acuerdo con la modalidad de codificación cuando el indicador de nivel de CU es 1. Además, el indicador de nivel de CU se puede señalar solo cuando al menos uno de los indicadores de bloque codificados en tres componentes de color (es decir, cbf_luma, cbf_cb y cbf_cr) es igual a 1.

15 **[0096]** Las técnicas de la divulgación actual también pueden incluir un procedimiento de obtención y señalización de matriz de transformación, tal como en el nivel de trama. En un ejemplo, para cada trama, se puede obtener una matriz de transformación en función de las estadísticas de los componentes de color en la imagen original. La matriz de transformación puede codificarse en un flujo de bits y transmitirse directamente al decodificador. Alternativamente, se puede aplicar la predicción de la matriz de transformación entre tramas codificadas consecutivamente. Por ejemplo, se puede transmitir un indicador para indicar si la matriz es la misma que la usada en la trama anterior. Alternativamente, una matriz de transformación podría obtenerse basándose en un conjunto de tramas originales y señalizarse en el flujo de bits.

25 **[0097]** En otro ejemplo, se obtienen y se señalizan un conjunto de matrices de transformación en la sintaxis de alto nivel, tal como el PPS. En un ejemplo, la imagen se clasifica en múltiples regiones según las características y, para cada región, se obtiene una matriz de transformación. Las matrices obtenidas se pueden codificar y señalar en la sintaxis de alto nivel. El índice de la matriz de transformación seleccionada se señala adicionalmente en el nivel de CU / TU / PU. Alternativamente, el índice de la matriz de transformación seleccionada se señala en la cabecera de fragmento o para cada región de mosaico. Alternativamente, la matriz de transformación se puede obtener en la cabecera de fragmento o para cada región de mosaico.

35 **[0098]** Cuando las profundidades en bits de los componentes de luma y croma son diferentes, la profundidad de bits de todos los componentes de color puede modificarse primero para que sean iguales, entonces la transformación de color puede aplicarse posteriormente. Alternativamente, el flujo de bits puede ser conforme a una restricción de que, cuando las profundidades en bits de los componentes de luma y croma son diferentes en el flujo de bits codificado, la transformación de color se desactivará (es decir, indicador_transform_color_habilitada igual a 0).

40 **[0099]** Se pueden aplicar diferentes QP a las CU / los bloques que están codificados con o sin transformación de color, y / o la profundidad en bits utilizada en la cuantización y el proceso de transformación puede modificarse debido a dicha transformación de color no normalizada. En un ejemplo, cuando se usa la transformación YCoCg modificada anteriormente, las siguientes etapas (es decir, el proceso de ajuste a escala, transformación y construcción de formación antes del proceso de filtro de desbloqueo) se modifican adicionalmente en función del *indicador_transform_color*.

45 **[0100]** En un ejemplo, cuando se utiliza la transformación YCoCg modificada, como se ha descrito anteriormente, los correspondientes procesos de obtención de parámetros de cuantización y procesos de transformación inversa se modifican en consecuencia. Las entradas para este proceso pueden incluir una ubicación de luminancia (xCb, yCb) que especifique la muestra superior izquierda del bloque de codificación de luminancia actual en relación con la muestra de luminancia superior izquierda de la imagen actual. En este proceso, se pueden obtener la variable QpY, el parámetro de cuantización de luma Qp'Y y los parámetros de cuantización de croma Qp'Cb y Qp'Cr. La ubicación de luma (xQg, yQg) puede especificar la muestra de luma superior izquierda del grupo de cuantización actual en relación con la muestra de luma superior izquierda de la imagen actual. Las posiciones horizontal y vertical xQg y yQg se pueden fijar iguales a $xCb - (xCb \& ((1 < \llcorner \text{Log2MinCuQpDeltaTamaño}) - 1))$ y a $yCb - (yCb \& ((1 < \llcorner \text{Log2MinCuQpDeltaTamaño}) - 1))$, respectivamente. El tamaño de luma de un grupo de cuantización, $\text{Log2MinCuQpDeltaTamaño}$, puede determinar el tamaño de luma del área más pequeña dentro de un bloque arbolado de codificación que comparte la misma qPY_PRED.

[0101] La variable QpY se puede obtener de la siguiente manera:

60
$$QpY = ((qPY_PRED + CuQpDeltaVal + 52 + 2 * QpBdDesplazamientoY) \% (52 + QpBdDesplazamientoY)) - QpBdDesplazamientoY$$

[0102] El parámetro de cuantización de luma Qp'Y se puede obtener de la siguiente manera:

65
$$Qp'Y = QpY + QpBdDesplazamientoY$$

[0103] Cuando el elemento sintáctico TipoFormaciónCroma no es igual a 0, se puede aplicar lo siguiente:

Las variables qPiCb y qPiCr se pueden obtener de la siguiente manera:

$$qPiCb = \text{Clip3}(-QpBdDesplazamientoC, 57, QpY + \text{desplazamiento_pps_cb_qp} + \text{desplazamiento_fragmento_cb_qp} + CuQpAdjValCb)$$

$$qPiCr = \text{Clip3}(-QpBdDesplazamientoC, 57, QpY + \text{desplazamiento_pps_cr_qp} + \text{desplazamiento_fragmento_cr_qp} + CuQpAdjValCr)$$

[0104] Si el elemento sintáctico TipoFormaciónCroma es igual a 1, las variables qPCb y qPCr se pueden fijar iguales al valor de QpC, como se especifica en la tabla a continuación basándose en el índice qPi igual a qPiCb y qPiCr, respectivamente. De lo contrario, las variables qPCb y qPCR se fijan iguales a Min (QPI, 51), basándose en el índice de QPI igual a qPiCb y qPiCr, respectivamente.

Tabla 2

qPi	< 30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	> 43
Qpc	= qPi	29	30	31	32	33	33	34	34	35	35	36	36	37	37	= qPi - 6

[0105] Los parámetros de cuantización de croma para los componentes Cb y Cr, Qp'Cb y Qp'Cr, se pueden obtener de la siguiente manera:

$$Qp'Cb = qPCb + QpBdDesplazamientoC$$

$$Qp'Cr = qPCr + QpBdDesplazamientoC$$

[0106] Cuando indicador_transform_color es igual a 1, puede valer lo siguiente:

$$Qp'Y = (Qp'Y + 8)$$

Cuando TipoFormaciónCroma ≠ 0, Qp'Cb = (Qp'cb + 8) y Qp'Cr = (Qp'cr + 8)

$$\text{ProfundidadBitsY} = (\text{ProfundidadBitsY} + 2)$$

$$\text{ProfundidadBitsC} = (\text{ProfundidadBitsC} + 2)$$

[0107] Las entradas para este proceso pueden incluir una ubicación de luminancia (xTbY, yTbY) que especifique la muestra superior izquierda del bloque de transformación de luminancia actual en relación con la muestra de luminancia superior izquierda de la imagen actual, una variable nTbS que especifica el tamaño del bloque de transformación actual, una variable cldx que especifica el componente de color del bloque actual y una variable qp que especifica el parámetro de cuantización. La salida de este proceso puede incluir la matriz d de tamaño (nTbS) x (nTbS) de coeficientes de transformación ajustados a escala con elementos d [x] [y].

[0108] Las variables log2RangoTransformación, desplazamientoBd, minCoef y maxCoef se pueden obtener de la siguiente manera:

Cuando indicador_transform_color = 1, vale lo siguiente:

$$\text{CoefCTMinY} = - (1 \ll (\text{indicador_procesamiento_precisión_extendida} ? \text{Max}(15, \text{ProfundidadBitsY} + 6) : 15))$$

$$\text{CoefCTMinC} = - (1 \ll (\text{indicador_procesamiento_precisión_extendida} ? \text{Max}(15, \text{ProfundidadBitsC} + 6) : 15))$$

$$\text{CoefCTMaxY} = (1 \ll (\text{indicador_procesamiento_precisión_extendida} ? \text{Max}(15, \text{ProfundidadBitsY} + 6) : 15)) - 1$$

$$\text{CoefCTMaxC} = (1 \ll (\text{indicador_procesamiento_precisión_extendida} ? \text{Max}(15, \text{ProfundidadBitsC} + 6) : 15)) - 1$$

si $cIdx = 0$,

5 $\log2RangoTransform = indicador_procesamiento_precisi3n_extendida ?$
 $Max(15, ProfundidadBitsY + 6): 15$

$desplazamientoBd = ProfundidadBitsY + Log2(nTbS) + 10 - \log2RangoTransform$

10 $coefMin = (indicador_transform_color ? CoefCTMinY: CoefMinY)$

$coefMax = (indicador_transform_color ? CoefCTMaxY: CoefMaxY)$

De otra manera,

15 $\log2RangoTransform = indicador_procesamiento_precisi3n_extendida ?$
 $Max(15, ProfundidadBitsC + 6): 15$

$desplazamientoBd = ProfundidadBitsC + Log2(nTbS) + 10 - \log2RangoTransform$

20 $coefMin = (indicador_transform_color ? CoefCTMinC: CoefMinC)$

$coefMax = (indicador_transform_color ? CoefCTMaxC: CoefMaxC)$

25 **[0109]** La lista $escalaNivel []$ se puede especificar como $escalaNivel [k] = \{40, 45, 51, 57, 64, 72\}$ con $k = 0..5$.

[0110] Para la obtenci3n de los coeficientes de transformaci3n ajustados a escala $d [x] [y]$, con $x = 0..nTbS - 1$, $y = 0..nTbS - 1$, puede valer lo siguiente:

30 El factor de ajuste a escala $m [x] [y]$ se puede obtener de la siguiente manera:

Si una o m3s de las siguientes condiciones son verdaderas, $m [x] [y]$ se fija igual a 16:

$indicador_lista_ajuste_a_escala_habilitada = 0$.

35 $indicador_omisi3n_transform [xTbY] [yTbY] = 1$ y $nTbS > 4$.

De lo contrario, vale lo siguiente:

40 $m [x] [y] = FactorAjusteAEscala [identificadorTama3o] [identificadorMatriz] [x] [y] \quad (8-283)$

45 **[0111]** Cuando se especifica el elemento sint3ctico $identificadorTama3o$ para el tama3o de la matriz de cuantizaci3n igual a $(nTbS) \times (nTbS)$ y el $identificadorMatriz$ se especifica en la Tabla 7-4 para el $identificadorTama3o$, $ModalidadPredCu [xTbY] [yTbY]$ y $cIdx$, respectivamente, entonces el coeficiente de transformaci3n ajustado a escala $d [x] [y]$ se puede obtener de la siguiente manera:

50 $d [x] [y] = Clip3 (coefMin, coefMax, ((NivelCoefTrans [xTbY] [yTbY] [cIdx]$
 $[x] [y] * m [x] [y] * escalaNivel [qP \% 6] \ll (qP / 6)) + (1 \ll (desplazamientoBd - 1))) \gg$
 $desplazamientoBd)$

[0112] Cuando $indicador_transform_color$ es igual a 1, puede valer lo siguiente:

$ProfundidadBitsY = ProfundidadBitsY - 2$

55 $ProfundidadBitsC = ProfundidadBitsC - 2$

60 **[0113]** En un ejemplo, cuando la transformaci3n de color-espacio se aplica a las modalidades intra, el residuo se obtiene en el dominio de color-espacio convertido. Es decir, antes de invocar el proceso de predicci3n intramuestral, los p3xeles adyacentes de la CU actual se pueden convertir primero a otro color-espacio sub3ptimo con una transformaci3n YCoCg o YCoCg-R directa modificada. Los p3xeles vecinos modificados se pueden usar entonces para obtener el predictor de la CU actual. El residuo (es decir, el error de predicci3n) se obtiene de la CU actual y de los p3xeles vecinos de la CU actual en el dominio sub3ptimo. El residuo se codifica de la misma manera que el flujo de

codificación convencional, tal como la predicción residual, la transformación, la cuantización y la codificación por entropía entre componentes. Después de la codificación por entropía, los coeficientes de transformación pueden modificarse adicionalmente con la transformación YCoCg inversa o YCoCg-R. Los coeficientes de transformación modificados se pueden agregar luego al predictor para obtener los píxeles reconstruidos de la CU actual. Después de invocar el proceso de reconstrucción, la transformación YCoCg o YCoCg-R inversa se puede aplicar a los píxeles vecinos modificados. Por lo tanto, en el lado del decodificador, antes de invocar el proceso de reconstrucción, se puede aplicar un proceso adicional en el que la transformación inversa de color-espacio se aplica a los coeficientes de transformación obtenidos.

[0114] Cuando la transformación de color-espacio se aplica a las modalidades intra, en lugar de generar el residuo restando el predictor al bloque actual en el dominio de color convertido, el residuo puede generarse usando los píxeles antes de la transformación de color-espacio, seguido por la transformación de color-espacio. En este caso, las modalidades intra de luma y croma se pueden fijar en la modalidad idéntica. Alternativamente, la señalización de la modalidad de croma puede omitirse. Alternativamente, el indicador de transformación de color se puede señalar solo cuando las modalidades intra de luma y croma son las mismas.

[0115] Como se ha descrito anteriormente, en algunos ejemplos, los procedimientos de codificación sin pérdidas y con pérdidas pueden compartir la misma matriz, por ejemplo, YCoCg-R. Adicionalmente, o alternativamente, cuando la transformación YCoCg-R se aplica a la codificación con pérdidas, la profundidad de bits de los componentes de croma de los bloques con transformación de color puede incrementarse en 1 en comparación con los bloques sin transformación de color. El QP del componente de luma de los bloques con transformación de color puede ser igual al de los bloques sin transformación de color menos 4. En un ejemplo, los dos QP de componentes de croma de bloques con transformación de color pueden ser iguales al QP de componente de croma / luma de bloques sin transformación de color más 2. En un ejemplo, los QP de componentes de croma Cg de bloques con transformación de color pueden ser iguales al QP de componente de luma / croma de bloques sin transformación de color más 2. Los QP de componentes de croma Co de bloques con transformación de color pueden ser iguales al QP del componente de luma / croma de bloques sin transformación de color más 3.

[0116] En tales ejemplos, cuando YCoCg-R se usa tanto para la modalidad con pérdidas como para la modalidad sin pérdidas, si el bloque actual está codificado con pérdidas, la profundidad en bits de luma y croma puede ser la misma para aquellos bloques codificados sin habilitar la transformación de color adaptativa. En este caso, cuando se aplica la codificación con pérdidas, después de la transformación directa de YCoCg-R, los componentes Co y Cg pueden ajustarse escala mediante el desplazamiento a la derecha en N bits para reducir la profundidad en bits aumentada debido a la transformación directa YCoCg-R y hacer que la precisión de la profundidad en bits sea la misma que la precisión de la profundidad en bits del componente Y. Además, antes de realizar la transformación YCoCg-R inversa, los componentes Co y Cg pueden modificarse con el desplazamiento a la izquierda de N bits. En un ejemplo, N es igual a 1.

[0117] En un ejemplo, se puede suponer que la profundidad en bits de entrada de los componentes luma y croma es la misma. Para la modalidad de codificación con pérdidas, las operaciones de desplazamiento se pueden aplicar a los dos componentes de croma (es decir, Cg, Co) después de la transformación directa, y antes de la transformación inversa. En un ejemplo, el siguiente proceso se aplica en orden:

1. la YCoCg-R directa no se cambia:

$$Co = R - B$$

$$t = B + [Co/2]$$

$$Cg = G - t$$

$$Y = t + [Cg/2]$$

2. Si el bloque actual está codificado en modalidad con pérdidas, se puede aplicar además lo siguiente:

$$Cg = Cg \gg 1;$$

$$Co = Co \gg 1;$$

[0118] En otro ejemplo, un desplazamiento puede considerarse en el proceso de desplazamiento a la derecha. Por ejemplo, las ecuaciones resaltadas anteriormente podrían reemplazarse por:

$$C_g = (C_g + 1) \gg 1;$$

$$C_o = (C_o + 1) \gg 1;$$

[0119] Antes de invocar la transformación YCoCg-R inversa, se puede aplicar lo siguiente:

5

Si el bloque actual está codificado en modalidad con pérdidas, puede valer lo siguiente:

$$C_g = C_g \ll 1;$$

$$C_o = C_o \ll 1;$$

10 Y la YCoCg-R inversa puede permanecer sin cambios:

$$t = Y - \lfloor C_g/2 \rfloor$$

$$G = C_g + t$$

$$B = t - \lfloor C_o/2 \rfloor$$

$$R = B + C_o$$

[0120] En tales ejemplos, se puede invocar un proceso de modificación residual para bloques de transformación utilizando la transformación de color adaptativa cuando el TipoFormaciónCroma es igual a 3. Las entradas para este proceso pueden incluir una variable tamañoBlq que especifica el tamaño del bloque, una formación de tamaño (tamañoBlq) x (tamañoBlq) de muestras de luma residual rY con elementos rY [x] [y], una matriz de tamaño (tamañoBlq) x (tamañoBlq) de muestras de croma residual rCb con elementos rCb [x] [y], y una matriz de tamaño (tamañoBlq) x (tamañoBlq) de muestras de croma residual rCr con elementos rCr [x] [y]. Las salidas de este proceso pueden incluir una matriz modificada rY de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de luma, una matriz modificada rCb de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma y una matriz modificada rCr de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma.

15

20

[0121] Las matrices de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales rY, rCb y rCr con x = 0..tamañoBlq - 1, y = 0..tamañoBlq - 1 pueden modificarse de la siguiente manera:

25

Si indicador_omisión_transcuant_cu = 0, rCb[x] [y] = rCb[x] [y] << 1 y rCr[x] [y] =

$$rCr[x][y] \ll 1$$

30

$$tmp = rY[x][y] - (rCb[x][y] \gg 1)$$

$$rY[x][y] = tmp + rCb[x][y]$$

$$rCb[x][y] = tmp - (rCr[x][y] \gg 1)$$

$$rCr[x][y] = rCb[x][y] + rCr[x][y]$$

35

[0122] Alternativamente, 'si indicador_omisión_transcuant_cu es igual a 0, rCb[x] [y] = rCb[x] [y] << 1 y rCr[x] [y] = rCr[x] [y] << 1' podría reemplazarse por 'rCb[x] [y] = rCb[x] [y] << (1- indicador_omisión_transcuant_cu) y rCr[x] [y] = rCr[x] [y] << (1- indicador_omisión_transcuant_cu)'.

[0123] En otro ejemplo de un procedimiento de codificación con pérdidas y un procedimiento de codificación sin pérdidas que comparten una matriz de transformación de color, la profundidad de bits de entrada de los componentes de luma y croma puede ser diferente. En un ejemplo de este tipo, cuando las profundidades en bits de entrada de los componentes de luma y croma son diferentes y YCoCg-R se utiliza para ambas modalidades de codificación con pérdidas y sin pérdidas, antes de la transformación inversa, al menos uno de los valores del componente Y, Co, Cg puede ser desplazado primero en N bits para hacer que los tres componentes tengan la misma precisión de profundidad en bits. Además, se puede desplazar un bit más a las dos componentes de croma (es decir, Co y Cg). Al menos uno de los valores de predicción de los tres componentes puede modificarse para que tenga la misma precisión de profundidad en bits antes de invocar la transformación directa YCoCg-R. Y después de la transformación directa, los dos componentes de croma se pueden desplazar a la derecha en 1 bit. Para las modalidades de codificación sin pérdidas, la diferente profundidad de en bits puede no ser considerada y, por lo tanto, la transformación de color-espacio adaptativa puede estar inhabilitada. En un ejemplo, el flujo de bits puede incluir una restricción de que, cuando la profundidad en bits de los componentes de luma y croma son diferentes en el flujo de bits codificado, la transformación de color se desactivará (es decir, indicador_transform_color_habilitada igual a 0).

[0124] En este ejemplo, se puede invocar un proceso de modificación residual para los bloques de transformación utilizando la transformación de color adaptativa cuando el TipoFormaciónCroma es igual a 3. Las entradas para este proceso pueden incluir una variable tamañoBlq que especifica el tamaño del bloque, una matriz de muestras de luma residual rY de tamaño (tamañoBlq) x (tamañoBlq) con elementos rY [x] [y], una matriz de muestras de croma residual rCb de tamaño (tamañoBlq) x (tamañoBlq) con elementos rCb [x] [y] y una matriz de muestras de croma residual rCr de tamaño (tamañoBlq) x (tamañoBlq) con elementos rCr [x] [y]. Las salidas de este proceso pueden incluir una matriz modificada rY de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de luma, una matriz modificada rCb de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma y una matriz modificada rCr de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma.

[0125] Las variables deltaBDY y deltaBDC se pueden obtener de la siguiente manera:

$$\text{MaxProfundidadBits} = \text{Max} (\text{ProfundidadBitsY}, \text{ProfundidadBitsC})$$

$$\text{deltaBDY} = \text{indicador_omisión_transcuant_cu} ? 0 : \text{MaxProfundidadBits} - \text{ProfundidadBitsY}$$

$$\text{deltaBDC} = \text{indicador_omisión_transcuant_cu} ? 0 : \text{MaxProfundidadBits} - \text{ProfundidadBitsC}$$

$$\text{OY} = \text{indicador_omisión_transcuant_cu} (\text{MaxProfundidadBits} == \text{ProfundidadBitsY}) ? 0 : 1 \ll (\text{deltaBDY} - 1)$$

$$\text{OC} = \text{indicador_omisión_transcuant_cu} (\text{MaxProfundidadBits} == \text{ProfundidadBitsC}) ? 0 : 1 \ll (\text{deltaBDC} - 1)$$

[0126] Las matrices de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales rY, rCb y rCr con x = 0..tamañoBlq - 1, y = 0..tamañoBlq - 1 pueden modificarse de la siguiente manera:

Las muestras residuales rY[x] [y], rCb[x] [y] y rCr[x] [y] se pueden modificar de la siguiente manera:

$$rY[x] [y] = rY[x] [y] \ll \text{deltaBDY}$$

$$rCb[x] [y] = rCb[x] [y] \ll (\text{deltaBDC} + 1 - \text{indicador_omisión_transcuant_cu})$$

$$rCr[x] [y] = rCr[x] [y] \ll (\text{deltaBDC} + 1 - \text{indicador_omisión_transcuant_cu})$$

$$\text{tmp} = rY[x][y] - (rCb[x][y] \gg 1)$$

$$rY[x][y] = \text{tmp} + rCb[x][y]$$

$$rCb[x][y] = \text{tmp} - (rCr[x][y] \gg 1)$$

$$rCr[x][y] = rCb[x][y] + rCr[x][y]$$

$$rY[x][y] = (rY[x][y] + OY) \gg \text{deltaBDY}$$

$$rCb[x][y] = (rCb[x][y] + Oc) \gg \text{deltaBDC}$$

$$rCr[x][y] = (rCr[x][y] + Oc) \gg \text{deltaBDC}$$

[0127] En los ejemplos anteriores, la función Max se puede usar para obtener el valor más grande entre dos variables. Alternativamente, al menos uno de los desplazamientos de redondeo O_Y y O_C puede ser igual a 0.

5 **[0128]** Alternativamente, incluso para la codificación sin pérdidas, cuando la profundidad en bits de los componentes luma y croma son diferentes, los componentes pueden desplazarse con la misma precisión de profundidad en bits, si es necesario. En el lado del codificador, después de la transformación directa, es posible que no haya necesidad de volver a la precisión original de profundidad en bits. En otras palabras, el residuo codificado de los tres componentes puede ser de la misma profundidad en bits para asegurarse de que sea una codificación sin pérdidas. A diferencia de la codificación con pérdidas, las entradas pueden no modificarse antes de invocar la transformación inversa YCoCg-R para la codificación sin pérdidas. Es posible que aún se necesite el desplazamiento a la derecha para asegurarse de que las salidas estén en la misma profundidad en bits que las entradas originales.

15 **[0129]** En este ejemplo, se puede invocar un proceso de modificación residual para los bloques de transformación utilizando la transformación de color adaptativa cuando el TipoFormaciónCroma es igual a 3. Las entradas para este proceso pueden incluir una variable tamañoBlq que especifica el tamaño del bloque, una matriz de muestras de luma residual rY de tamaño (tamañoBlq) x (tamañoBlq) con elementos $rY[x][y]$, una matriz de muestras de croma residual rCb de tamaño (tamañoBlq) x (tamañoBlq) con elementos $rCb[x][y]$ y una matriz de muestras de croma residual rCr de tamaño (tamañoBlq) x (tamañoBlq) con elementos $rCr[x][y]$. Las salidas de este proceso pueden incluir una matriz modificada r_Y de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de luma, una matriz modificada r_{cb} de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma y una matriz modificada r_{cr} de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales de croma.

25 **[0130]** Las variables deltaBD_Y y deltaBD_C se pueden obtener de la siguiente manera:

$$\text{MaxProfundidadBits} = \text{Max}(\text{ProfundidadBits}_Y, \text{ProfundidadBits}_C)$$

$$\text{deltaBD}_Y = \text{MaxProfundidadBits} - \text{ProfundidadBits}_Y$$

$$\text{deltaBD}_C = \text{MaxProfundidadBits} - \text{ProfundidadBits}_C$$

$$OY = (\text{MaxProfundidadBits} == \text{ProfundidadBits}_Y) ? 0 : 1 \ll (\text{deltaBD}_Y - 1)$$

$$OC = (\text{MaxProfundidadBits} == \text{ProfundidadBits}_C) ? 0 : 1 \ll (\text{deltaBD}_C - 1)$$

[0131] Las matrices de tamaño (tamañoBlq) x (tamañoBlq) de muestras residuales r_Y , r_{cb} y r_{cr} con $x = 0..tamañoBlq - 1$, $y = 0..tamañoBlq - 1$ pueden modificarse de la siguiente manera:

40 Cuando el indicador_omisión_transcuant_cu es igual a 0, las muestras residuales $rY[x][y]$, $rCb[x][y]$ y $rCr[x][y]$ se pueden modificar de la siguiente manera:

$$rY[x][y] = rY[x][y] \ll \text{deltaBD}_Y$$

$$45 \quad rCb[x][y] = rCb[x][y] \ll (\text{deltaBD}_C + 1 - \text{indicador_omisión_transcuant_cu})$$

$$rCr[x][y] = rCr[x][y] \ll (\text{deltaBD}_C + 1 - \text{indicador_omisión_transcuant_cu})$$

$$\text{tmp} = rY[x][y] - (rCb[x][y] \gg 1)$$

$$rY[x][y] = \text{tmp} + rCb[x][y]$$

$$rCb[x][y] = \text{tmp} - (rCr[x][y] \gg 1)$$

$$rCr[x][y] = rCb[x][y] + rCr[x][y]$$

$$rY[x][y] = (rY[x][y] + OY) \gg \text{deltaBDY}$$

$$rCb[x][y] = (rCb[x][y] + Oc) \gg \text{deltaBDC}$$

$$rCr[x][y] = (rCr[x][y] + Oc) \gg \text{deltaBDC}$$

[0132] Alternativamente, uno entre O_Y y O_C puede ser igual a 0.

5 [0133] Además, el codificador de vídeo 20 puede enviar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en tramas y datos sintácticos basados en GOP, al decodificador de vídeo 30, por ejemplo, en una cabecera de trama, una cabecera de bloque, una cabecera de fragmento o una cabecera de GOP. Los datos sintácticos de GOP pueden describir un cierto número de tramas en el respectivo GOP, y los datos sintácticos de trama pueden indicar una modalidad de codificación/predicción utilizada para codificar la trama correspondiente.

10 [0134] Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede implementarse como cualquiera entre una amplia variedad de circuitos codificadores o decodificadores adecuados, según corresponda, 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), circuitos de lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede estar incluido en uno o más codificadores o decodificadores, cada uno de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) de vídeo combinado. Un dispositivo que incluye el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

15 [0135] La figura 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar técnicas para codificar bloques de vídeo usando un proceso de conversión de color-espacio. El codificador de vídeo 20 puede realizar intracodificación e intercodificación de bloques de vídeo dentro de fragmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo determinada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. La intramodalidad (modalidad I) puede referirse a cualquiera de varias modalidades de codificación de base espacial. Las intermodalidades, tales como la predicción unidireccional (modalidad P) o la bipredicción (modalidad B), pueden referirse a cualquiera de varias modalidades de codificación de base temporal.

20 [0136] Como se muestra en la figura 2, el codificador de vídeo 20 recibe un bloque de vídeo actual dentro de una trama de vídeo a codificar. En el ejemplo de la figura 2, el codificador de vídeo 20 incluye una unidad de selección de modalidad 40, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformación 52, una unidad de cuantización 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modalidad 40 incluye una unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de intrapredicción 46 y una unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además la unidad de cuantización inversa 58, la unidad de transformación inversa 60 y el sumador 62. También puede incluirse un filtro de desbloqueo (no mostrado en la figura 2) para filtrar fronteras de bloques, a fin de 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 adicionales (en bucle o post-bucle), además del filtro de desbloqueo. Dichos filtros no se muestran por razones de brevedad pero, si se desea, pueden filtrar la salida del sumador 50 (tal como un filtro en bucle).

35 [0137] Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un fragmento de vídeo a codificar. La trama o el fragmento pueden dividirse en múltiples bloques de vídeo. La unidad de estimación de

movimiento 42 y la unidad de compensación de movimiento 44 llevan a cabo la codificación interpredictiva del bloque de vídeo recibido con respecto a uno o más bloques en una o más tramas de referencia, para proporcionar la predicción temporal. La unidad de intrapredicción 46, como alternativa, puede llevar a cabo la codificación intrapredictiva del bloque de vídeo recibido, con respecto a uno o más bloques contiguos en la misma trama o fragmento que el bloque a codificar, para proporcionar predicción espacial. El codificador de vídeo 20 puede llevar a cabo múltiples pases de codificación, por ejemplo, para seleccionar una modalidad de codificación adecuada para cada bloque de datos de vídeo.

[0138] Además, la unidad de división 48 puede dividir bloques de datos de vídeo en subbloques, basándose en la evaluación de los anteriores esquemas de división en los anteriores pases de codificación. Por ejemplo, la unidad de división 48 puede dividir inicialmente una trama o un fragmento en varias LCU, y dividir cada una de las LCU en varias sub-CU, basándose en un análisis de velocidad-distorsión (por ejemplo, optimización de velocidad-distorsión). La unidad de selección de modalidad 40 puede producir además una estructura de datos de árbol cuádruple, indicativa de la división de una LCU en varias sub-CU. Las CU de nodos de hojas del árbol cuádruple pueden incluir una o más PU y una o más TU.

[0139] La unidad de selección de modalidad 40 puede seleccionar una de las modalidades de codificación, intra o inter, por ejemplo, basándose en los resultados de errores, y proporciona el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloques residuales, y al sumador 62 para reconstruir el bloque codificado para su uso como una trama de referencia. La unidad de selección de modalidad 40 proporciona además elementos sintácticos, tales como vectores de movimiento, indicadores de intramodalidad, información de división y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

[0140] La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual, con respecto a un bloque predictivo dentro de una trama de referencia (u otra unidad codificada), con respecto al bloque actual que se está codificando dentro de la trama actual (u otra unidad codificada). Un bloque predictivo es un bloque que se revela como estrechamente coincidente con el bloque a codificar, en lo que respecta a la diferencia de píxeles, lo cual puede determinarse mediante la suma de una diferencia absoluta (SAD), una suma de diferencia de cuadrados (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxeles de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento en relación con las posiciones de píxeles completos y las posiciones fraccionarias de píxeles, y emitir un vector de movimiento con una precisión de píxel fraccionaria.

[0141] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento intercodificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse 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. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación por entropía 56 y a la unidad de compensación de movimiento 44.

[0142] La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar capturar o generar el bloque predictivo basándose en el vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden integrarse funcionalmente, en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que se está codificando, generando valores de diferencia de píxel, como se expone posteriormente. En general, la unidad de estimación de movimiento 42 lleva a cabo la estimación de movimiento con respecto a los componentes de luma, y la unidad de compensación de movimiento 44 utiliza los vectores de movimiento calculados basándose en los componentes de luma, tanto para los componentes de croma como para los componentes de luma. La unidad de selección de modalidad 40 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el decodificador de vídeo 30 al decodificar los bloques de vídeo del fragmento de vídeo.

[0143] La unidad de intrapredicción 46 puede intrapredicir un bloque actual, como alternativa a la interpredicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de intrapredicción 46 puede determinar una modalidad de

intrapredicción a utilizar para codificar un bloque actual. En algunos ejemplos, la unidad de intrapredicción 46 puede codificar un bloque actual usando varias modalidades de intrapredicción, por ejemplo, durante diferentes pases de codificación, y la unidad de intrapredicción 46 (o la unidad de selección de modalidad 40, en algunos ejemplos) puede seleccionar una modalidad de intrapredicción adecuada a utilizar a partir de las modalidades probadas.

[0144] Por ejemplo, la unidad de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para las diversas modalidades de intrapredicción probadas, y seleccionar la modalidad de intrapredicción que tenga las mejores características de velocidad-distorsión entre las modalidades probadas. El análisis de velocidad-distorsión determina en general una magnitud de distorsión (o de errores) entre un bloque codificado y un bloque original, no codificado, que se codificó para generar el bloque codificado, así como una velocidad binaria (es decir, un número de bits) usada para generar el bloque codificado. La unidad de intrapredicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modalidad de intrapredicción presenta el mejor valor de velocidad-distorsión para el bloque.

[0145] Después de seleccionar una modalidad de intrapredicción para un bloque, la unidad de intrapredicción 46 puede proporcionar información, indicativa de la modalidad de intrapredicción seleccionada para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indica la modalidad de intrapredicción seleccionada. 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 modalidades de intrapredicción y una pluralidad de tablas modificadas de índices de modalidades de intrapredicción (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones de la modalidad de intrapredicción más probable, una tabla de índices de modalidades de intrapredicción y una tabla modificada de índices de modalidades de intrapredicción, a utilizar para cada uno de los contextos.

[0146] El codificador de vídeo 20 forma un bloque de vídeo residual restando los datos de predicción de la unidad de selección de modalidad 40 al bloque de vídeo original que se está codificando. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformaciones 52 aplica una transformación, tal como una transformación discreta de coseno (DCT) o una transformación conceptualmente similar, al bloque residual, generando un bloque de vídeo que comprende valores residuales de coeficientes de transformación. La unidad de procesamiento de transformaciones 52 puede llevar a cabo otras transformaciones que son conceptualmente similares a la DCT. También podrían usarse transformaciones de ondículas, transformaciones enteras, transformaciones de subbandas u otros tipos de transformaciones. En cualquier caso, la unidad de procesamiento de transformaciones 52 aplica la transformación al bloque residual, generando un bloque de coeficientes de transformación residuales. La transformación puede convertir la información residual, desde un dominio de valores de píxel a un dominio de transformaciones, tal como un dominio de frecuencia. La unidad de procesamiento de transformaciones 52 puede enviar los coeficientes de transformación resultantes a la unidad de cuantización 54. La unidad de cuantización 54 cuantiza los coeficientes de transformación para reducir adicionalmente la velocidad de transmisión de bits. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes. El grado de cuantización puede modificarse ajustando un parámetro de cuantización. En algunos ejemplos, la unidad de cuantización 54 puede realizar, a continuación, un recorrido de la matriz que incluye los coeficientes de transformación cuantizados. De forma alternativa, la unidad de codificación por entropía 56 puede llevar a cabo el recorrido.

[0147] Tras la cuantización, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformación cuantizados. Por ejemplo, la unidad de codificación por entropía 56 puede llevar a cabo la codificación de longitud variable adaptable al contexto (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación aritmética binaria adaptable al contexto y basada en sintaxis (SBAC), la codificación por entropía mediante la división en intervalos de probabilidades (PIPE) u otra técnica de codificación por entropía. En el caso de la codificación por entropía basada en el contexto, el contexto puede basarse en bloques contiguos. Tras la codificación por entropía realizada por la unidad de codificación por entropía 56, el flujo de bits codificado puede transmitirse a otro dispositivo (por ejemplo, el decodificador de vídeo 30) o archivarse para su posterior transmisión o recuperación.

[0148] De acuerdo con las técnicas de esta divulgación, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede realizar una o más técnicas de la divulgación actual. Por ejemplo, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar una modalidad de codificación utilizada para codificar los datos de vídeo. La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. La unidad de codificación por entropía 56 puede determinar además un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además el proceso de transformación de color-espacio a los datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación. En las técnicas de esta divulgación, el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

[0149] La unidad de cuantización inversa 58 y la unidad de transformación inversa 60 aplican la cuantización inversa

y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de una de las tramas de la memoria de imágenes de referencia 64. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxeles fraccionarios para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento, generado por la unidad de compensación de movimiento 44, para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria de imágenes de referencia 64. El bloque de vídeo reconstruido puede ser utilizado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como un bloque de referencia para intercodificar un bloque en una trama de vídeo posterior.

[0150] De esta manera, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede realizar una o más técnicas de la divulgación actual. Por ejemplo, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede realizar una o más técnicas de la divulgación actual. Por ejemplo, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar una modalidad de codificación utilizada para codificar los datos de vídeo. La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. La unidad de codificación por entropía 56 puede determinar además un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además el proceso de transformación de color-espacio a los datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación. En las técnicas de esta divulgación, el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

[0151] En algunos ejemplos, la modalidad de codificación puede ser la modalidad de codificación con pérdidas. En tales ejemplos, el proceso de transformación de color-espacio puede ser una matriz YCoCg. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede modificar un componente Co y un componente CG de la matriz YCoCg con un desplazamiento de N bits a la izquierda antes de aplicar el proceso de transformación inversa de color-espacio en el bucle de decodificación del proceso de codificación. En algunos ejemplos, N puede ser igual a 1.

[0152] En otros ejemplos, la modalidad de codificación puede ser la modalidad de codificación sin pérdidas. En tales ejemplos, el proceso de transformación de color-espacio puede ser una matriz YCoCg-R. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aumentar la profundidad de bits de un componente Co y de un componente Cg de la matriz YCoCg-R en 1.

[0153] En cualquiera de los dos ejemplos anteriores, el proceso de transformación inversa de color-espacio puede ser un proceso de transformación inversa de color-espacio sin pérdidas. En tales ejemplos, el proceso de transformación inversa de color-espacio sin pérdidas puede comprender una matriz YCoCg-R.

[0154] En algunos ejemplos, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además usar la conversión de color-espacio para codificar los datos de vídeo. Al hacerlo, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar un valor de un elemento sintáctico de los datos de vídeo para indicar el uso de la conversión de color-espacio en respuesta a la determinación de usar la conversión de color-espacio para la unidad de codificación. En algunos ejemplos, el elemento sintáctico comprende un indicador de un bit. En algunos ejemplos, el elemento sintáctico se señala cuando la unidad de codificación se codifica utilizando una modalidad distinta a la modalidad de modulación de código de intrapulsos (IPCM). En otros ejemplos, el elemento sintáctico se señala solo cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación. En algunos ejemplos, un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio.

[0155] En algunos ejemplos, el elemento sintáctico puede no estar señalado. Por ejemplo, el elemento sintáctico puede no estar señalado cuando la unidad de codificación está intracodificada y cuando una modalidad de predicción de luma y una modalidad de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes. En otro ejemplo, el elemento sintáctico puede no ser señalado cuando la unidad de codificación está codificada con una modalidad de paleta.

[0156] En algunos ejemplos, el elemento sintáctico es un primer elemento sintáctico y los datos de vídeo son un primer conjunto de datos de vídeo. En tales ejemplos, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar un valor de un índice de paleta para un primer píxel en una segunda unidad de codificación de un segundo conjunto de datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además los valores de los índices de paleta para uno o más píxeles siguientes en un orden de recorrido inmediatamente posterior al primer píxel. La determinación de los valores de los índices de paleta puede comprender determinar los valores de los índices de paleta hasta que un píxel tenga un índice de paleta con un valor diferente al valor del índice de paleta para el primer píxel. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además una serie de valores de índice de paleta determinados para los siguientes píxeles.

Según el número de valores de índice de paleta, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede codificar el índice de paleta utilizando una primera modalidad o una segunda modalidad. Cuando el número de valores de índice de paleta es mayor que uno, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar un valor de un segundo elemento sintáctico para indicar que la primera modalidad se usó para codificar el índice de paleta y fijar un valor de un tercer elemento sintáctico igual al número de valores de índice de paleta. Cuando el número de valores de índice de paleta es menor o igual a uno, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar el valor del segundo elemento sintáctico para indicar que la segunda modalidad se usó para codificar el índice de paleta. En algunos ejemplos, la primera modalidad es una modalidad de racha, y la segunda modalidad es una modalidad de píxel.

[0157] En algunos ejemplos, la codificación del elemento sintáctico puede comprender que la unidad de codificación por entropía 56 del codificador de vídeo 20 codifique datos de un mensaje de información de mejora suplementaria (SEI) que indica si la unidad de codificación fue codificada utilizando la conversión de color-espacio.

[0158] La figura 3 es un diagrama de bloques que ilustra un ejemplo del decodificador de vídeo 30 que puede implementar técnicas para decodificar bloques de vídeo, algunos de los cuales fueron codificados usando un proceso de conversión de color-espacio. En el ejemplo de la figura 3, el decodificador de vídeo 30 incluye una unidad de decodificación por entropía 70, una unidad de compensación de movimiento 72, una unidad de intrapredicción 74, una unidad de cuantización inversa 76, una unidad de transformación inversa 78, una memoria de imágenes de referencia 82 y un sumador 80. En algunos ejemplos, el decodificador de vídeo 30 puede llevar a cabo un pase de decodificación generalmente recíproco al pase de codificación descrito con respecto al codificador de vídeo 20 (figura 2). La unidad de compensación de movimiento 72 puede generar datos de predicción basándose en vectores de movimiento recibidos desde la unidad de decodificación por entropía 70, mientras que la unidad de intrapredicción 74 puede generar datos de predicción basándose en indicadores de modalidad de intrapredicción recibidos desde la unidad de decodificación por entropía 70.

[0159] 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. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 decodifica por entropía el flujo de bits para generar coeficientes cuantizados, vectores de movimiento o indicadores de modalidad de intrapredicción y otros elementos sintácticos. La unidad de decodificación por entropía 70 remite los vectores de movimiento y otros elementos sintácticos a la unidad de compensación de movimiento 72. El decodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel del fragmento de vídeo y/o el nivel del bloque de vídeo.

[0160] De acuerdo con las técnicas de esta divulgación, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede realizar una o más técnicas de la divulgación actual. Por ejemplo, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede recibir un primer bloque codificado de datos de vídeo. El primer bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede recibir además un segundo bloque codificado de datos de vídeo. El segundo bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede avanzar un proceso de transformación inversa de color-espacio al primer bloque codificado de datos de vídeo. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar además el mismo proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo. En el lado del decodificador, puede que no sea necesario realizar un proceso de transformación directa de color-espacio, independientemente de la modalidad de codificación. En algunos ejemplos, el proceso de transformación inversa de color-espacio puede ser fijo.

[0161] Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), la unidad de intrapredicción 74 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual basándose en una modalidad de intrapredicción señalizada y en datos de bloques previamente decodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B, P o GPB), la unidad de compensación de movimiento 72 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual basándose en los vectores de movimiento y en otros elementos sintácticos recibidos desde la unidad de decodificación por entropía 70. Los bloques predictivos pueden generarse 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, lista 0 y lista 1, usando técnicas de construcción predeterminadas, basándose en imágenes de referencia almacenadas en la memoria de imágenes de referencia 82. La unidad de compensación de movimiento 72 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 se está decodificando. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar una modalidad de predicción (por ejemplo, intrapredicción o interpredicción) usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia del fragmento, vectores de movimiento para cada bloque de vídeo intercodificado del fragmento, el estado de interpredicción para cada bloque de vídeo intercodificado

del fragmento y otra información para decodificar los bloques de vídeo en el fragmento de vídeo actual.

5 **[0162]** La unidad de compensación de movimiento 72 también puede realizar la interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo, para calcular valores interpolados para píxeles fraccionarios de bloques de referencia. En este caso, la unidad de compensación de movimiento 72 puede determinar los filtros de interpolación utilizados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y utilizar los filtros de interpolación para generar bloques predictivos.

10 **[0163]** La unidad de cuantización inversa 76 cuantiza de manera inversa, es decir, descuantiza, los coeficientes de transformación cuantizados, proporcionados en el flujo de bits y decodificados por la unidad de decodificación por entropía 70. El proceso de cuantización inversa puede incluir el uso de un parámetro de cuantización QPy, calculado por el decodificador de vídeo 30 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.

15 **[0164]** La unidad de transformación inversa 78 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, a fin de producir bloques residuales en el dominio de píxeles.

20 **[0165]** Después de que la unidad de compensación de movimiento 72 genera el bloque predictivo para el bloque de vídeo actual basándose en los vectores de movimiento y otros elementos sintácticos, el decodificador de vídeo 30 forma un bloque de vídeo decodificado sumando los bloques residuales de la unidad de transformación inversa 78 y los correspondientes bloques predictivos generados por la unidad de compensación de movimiento 72. El sumador 80 representa el componente o los componentes que realizan 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. También pueden utilizarse otros filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) 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 determinada se almacenan a continuación en la memoria de imágenes de referencia 82, que almacena imágenes de referencia usadas para una posterior compensación de movimiento. La memoria de imágenes de referencia 82 almacena también vídeo decodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la figura 1.

35 **[0166]** De esta manera, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede realizar una o más técnicas de la divulgación actual. Por ejemplo, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede recibir un primer bloque codificado de datos de vídeo. El primer bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede recibir además un segundo bloque codificado de datos de vídeo. El segundo bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede avanzar un proceso de transformación inversa de color-espacio al primer bloque codificado de datos de vídeo. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar además el mismo proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo. En el lado del decodificador, puede que no sea necesario realizar un proceso de transformación directa de color-espacio, independientemente de la modalidad de codificación. En algunos ejemplos, el proceso de transformación inversa de color-espacio puede ser fijo. En algunos ejemplos, los bloques de datos de vídeo pueden ser unidades de codificación.

50 **[0167]** En algunos ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede modificar además uno o más componentes del primer bloque codificado de datos de vídeo con un desplazamiento a la derecha de N bits o un desplazamiento a la izquierda de N bits. En algunos ejemplos, N puede ser igual a 1. En algunos ejemplos, los uno o más componentes del primer bloque codificado de datos de vídeo pueden ser dos componentes de croma.

55 **[0168]** En algunos ejemplos, el primer proceso de transformación de color-espacio comprende una matriz YCoCg. En otros ejemplos, el segundo proceso de transformación de color-espacio comprende una matriz YCoCg-R. En cualquiera de los dos ejemplos, el proceso de transformación inversa de color-espacio puede comprender un proceso de transformación inversa de color-espacio sin pérdidas. En algunos ejemplos, el proceso de transformación inversa sin pérdidas de color-espacio comprende una matriz YCoCg-R.

60 **[0169]** En algunos ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar un elemento sintáctico de una unidad de codificación de datos de vídeo. El elemento sintáctico puede indicar si la unidad de codificación se codificó utilizando la conversión de color-espacio. En algunos ejemplos, el elemento sintáctico puede comprender un indicador de un bit. En algunos ejemplos, la unidad de codificación se codifica en una modalidad diferente a la modalidad de modulación de código de inimpulso (IPCM) y el elemento sintáctico se señala solo para unidades de codificación que usan una modalidad diferente a la modalidad de IPCM. En algunos ejemplos, el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión

de color-espacio cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación. En algunos ejemplos, un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar además si un valor del elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio. En respuesta a la determinación de que el elemento sintáctico indica que la unidad de codificación se codificó utilizando conversión de color-espacio, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar el proceso de transformación inversa de color-espacio.

[0170] En algunos ejemplos, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación. Por ejemplo, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación cuando la unidad de codificación está intracodificada y cuando una modalidad de predicción de luma y una modalidad de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes. En otro ejemplo, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación cuando la unidad de codificación se codifica con una modalidad de paleta. En estos ejemplos, el elemento sintáctico puede no estar presente en un flujo de bits recibido que comprende los datos de vídeo, y la decodificación del elemento sintáctico puede comprender deducir el valor del elemento sintáctico.

[0171] En algunos ejemplos, el elemento sintáctico es un primer elemento sintáctico y los datos de vídeo son un primer conjunto de datos de vídeo. En tales ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar además un valor de un segundo elemento sintáctico de un segundo conjunto de datos de vídeo. El segundo elemento sintáctico puede indicar si se usó una modalidad de paleta para codificar el segundo conjunto de datos de vídeo. En respuesta al segundo elemento sintáctico que indica que la modalidad de paleta se utilizó para codificar los datos de vídeo, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar el valor de un tercer elemento sintáctico. El tercer elemento sintáctico puede indicar si se utilizó una primera modalidad o una segunda modalidad para decodificar un índice de paleta para un píxel en la unidad de codificación. Basándose en el valor determinado del segundo elemento sintáctico, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar el índice de paleta utilizando la primera modalidad o la segunda modalidad. Cuando se utiliza la primera modalidad, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar un valor del índice de paleta, determinar un valor de un cuarto elemento sintáctico que indica un número de píxeles en un orden de recorrido que siga inmediatamente al píxel que se está decodificando actualmente, y duplicar el resultado de la determinación del valor del índice de paleta para los siguientes N píxeles en el orden de recorrido, con N igual al valor del cuarto elemento sintáctico. En algunos ejemplos, la primera modalidad es una modalidad de racha. Cuando se utiliza la segunda modalidad, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar el valor del índice de paleta y generar un valor de muestra de píxel para el píxel, donde el valor de muestra de píxel es igual al valor del índice de paleta. En algunos ejemplos, la segunda modalidad es una modalidad de píxel.

[0172] En algunos ejemplos, la decodificación del elemento sintáctico comprende la unidad de decodificación por entropía 70 del decodificador de vídeo 30, configurada para decodificar datos de un mensaje de información de mejora suplementaria (SEI) que indica si la unidad de codificación se codificó utilizando la conversión de color-espacio.

[0173] La figura 4 es un diagrama conceptual que ilustra las 35 modalidades de predicción de la HEVC de acuerdo con una o más técnicas de la divulgación actual. En la HEVC actual, para el componente de luma de cada Unidad de Predicción (PU), se utiliza un procedimiento de intrapredicción con 33 modalidades de predicción angular (indizadas de 2 a 34), la modalidad DC (indizada con 1) y la modalidad Plana (indizada con 0), como se describe con respecto a la figura 4.

[0174] Además de las 35 intramodalidades anteriores, la HEVC también emplea una modalidad más, denominada 'I-PCM'. En la modalidad I-PCM, la predicción, la transformación, la cuantización y la codificación por entropía se omiten mientras que las muestras de predicción sean codificadas mediante un número predefinido de bits. El propósito principal de la modalidad I-PCM es gestionar la situación cuando la señal no puede ser codificada eficazmente por otras modalidades.

[0175] La figura 5 es un diagrama conceptual que ilustra los candidatos de vectores de movimiento adyacentes espaciales para las modalidades de fusión y de predicción avanzada de vectores de movimiento (AMVP), de acuerdo con una o más técnicas de la divulgación actual. Como se describe con respecto a la figura 5, los candidatos de MV espacial se obtienen de los bloques vecinos que se muestran en la figura 5 para una PU específica (PU0), aunque los procedimientos que generan los candidatos a partir de los bloques difieren para las modalidades de fusión y de AMVP.

[0176] En la modalidad de fusión, se pueden obtener hasta cuatro candidatos de MV espaciales con los órdenes mostrados en la figura 5 (a) con números, y el orden es el siguiente: izquierda (0), arriba (1), arriba derecha (2), abajo izquierda (3) y arriba izquierda (4), como se muestra en la figura 5 (a).

[0177] En la modalidad de AMVP, los bloques vecinos se dividen en dos grupos: el grupo izquierdo consiste en los bloques 0 y 1, y el grupo superior consiste en los bloques 2, 3 y 4, como se muestra en la figura 5 (b). Para cada grupo,

el candidato potencial en un bloque vecino, que se refiere a la misma imagen de referencia que la indicada por el índice de referencia señalado, tiene la prioridad más alta para ser elegido para formar un candidato final del grupo. Es posible que todos los bloques vecinos no contengan un vector de movimiento que apunte a la misma imagen de referencia. Por lo tanto, si no se puede encontrar dicho candidato, el primer candidato disponible se ajustará a escala para formar el candidato final, por lo que las diferencias de distancia temporal se pueden compensar.

[0178] La figura 6 es un diagrama conceptual que ilustra un ejemplo de copia intrabloque (BC) de acuerdo con una o más técnicas de la divulgación actual. Como se describe con respecto a la figura 6, la Copia Intra-Bloque (BC) se ha incluido en RExt. Un ejemplo de Intra-BC se muestra como en la figura 6, en la que la CU actual se predice a partir de un bloque ya decodificado de la imagen / segmento actual. El tamaño actual del bloque de Intra-BC puede ser tan grande como un tamaño de CU, que varía de 8x8 a 64x64, aunque algunas aplicaciones, y restricciones adicionales, pueden valer adicionalmente.

[0179] La figura 7 es un diagrama conceptual que ilustra un ejemplo de un bloque de destino y una muestra de referencia para un intrabloque de tamaño 8x8, de acuerdo con una o más técnicas de la divulgación actual. Como se describe a continuación con respecto a la figura 7, una matriz de transformación se obtiene de los valores de muestra de referencia. Se utilizan diferentes muestras de referencia para el caso intra y el caso inter. Para el caso de intrabloque, el bloque de destino y las muestras de referencia se muestran en la figura 7. En esta figura, el bloque de destino consiste en muestras cuadriculadas de tamaño 8x8 y las referencias son muestras de rayas y puntos.

[0180] Para el caso de interbloque, las muestras de referencia para la obtención de la matriz son las mismas que para la compensación de movimiento. Para realizar la operación de desplazamiento, las muestras de referencia en el bloque de AMP se submuestran de tal manera que el número de muestras se convierta en la potencia de dos. Por ejemplo, el número de muestras de referencia en un bloque de tamaño 12x16 se reduce en 2/3.

[0181] De acuerdo con algunas técnicas de la presente divulgación, se puede aplicar el proceso de transformación de color-espacio. En tales ejemplos, no es necesario señalar si el proceso de transformación se invoca o no. Además, tanto el lado del codificador como el del decodificador pueden obtener la matriz de transformación con el mismo procedimiento para evitar la sobrecarga para señalar la matriz de transformación.

[0182] La figura 8 es un diagrama de flujo que ilustra un procedimiento a modo de ejemplo para codificar un bloque actual. El bloque actual puede comprender una CU actual o una parte de la CU actual. Aunque se ha descrito con respecto al codificador de vídeo 20 (figuras 1 y 2), debería entenderse que pueden configurarse otros dispositivos para realizar un procedimiento similar al de la figura 8.

[0183] En este ejemplo, el codificador de vídeo 20 inicialmente predice el bloque actual (150). Por ejemplo, el codificador de vídeo 20 puede calcular una o más unidades de predicción (PU) para el bloque actual. Posteriormente, el codificador de vídeo 20 puede calcular un bloque residual para el bloque actual, por ejemplo, para producir una unidad de transformación (TU) (152). Para calcular el bloque residual, el codificador de vídeo 20 puede calcular una diferencia entre el bloque original no codificado y el bloque predicho para el bloque actual. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar una modalidad de codificación usada para codificar los datos de vídeo (154). La modalidad de codificación puede ser una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas. La unidad de codificación por entropía 56 puede determinar además un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo (156). La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además el proceso de transformación de color-espacio a los datos de vídeo (158). Posteriormente, el codificador de vídeo 20 puede transformar y cuantizar los coeficientes del bloque residual (160). A continuación, el codificador de vídeo 20 puede recorrer los coeficientes de transformación cuantizados del bloque residual (162). Durante el recorrido, o tras el recorrido, el codificador de vídeo 20 puede codificar por entropía los coeficientes (164). Por ejemplo, el codificador de vídeo 20 puede codificar los coeficientes usando CAVLC o CABAC. Posteriormente, el codificador de vídeo 20 puede emitir los datos codificados por entropía del bloque (166). La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aplicar además un proceso de transformación inversa de color-espacio en un bucle de decodificación del proceso de codificación (168). En las técnicas de esta divulgación, el proceso de transformación inversa de color-espacio es independiente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas.

[0184] En algunos ejemplos, la modalidad de codificación puede ser la modalidad de codificación con pérdidas. En tales ejemplos, el proceso de transformación de color-espacio puede ser una matriz YCoCg. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede modificar un componente Co y un componente CG de la matriz YCoCg con un desplazamiento de N bits a la izquierda antes de aplicar el proceso de transformación inversa de color-espacio en el bucle de decodificación del proceso de codificación. En algunos ejemplos, N puede ser igual a 1.

[0185] En otros ejemplos, la modalidad de codificación puede ser la modalidad de codificación sin pérdidas. En tales ejemplos, el proceso de transformación de color-espacio puede ser una matriz YCoCg-R. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede aumentar la profundidad de bits de un componente Co y de un componente Cg de la matriz YCoCg-R en 1.

[0186] En cualquiera de los dos ejemplos anteriores, el proceso de transformación inversa de color-espacio puede ser un proceso de transformación inversa de color-espacio sin pérdidas. En tales ejemplos, el proceso de transformación inversa de color-espacio sin pérdidas puede comprender una matriz YCoCg-R.

5 **[0187]** En algunos ejemplos, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además usar la conversión de color-espacio para codificar los datos de vídeo. Al hacerlo, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar un valor de un elemento sintáctico de los datos de vídeo para indicar el uso de la conversión de color-espacio en respuesta a la determinación de usar la conversión de color-espacio para la unidad de codificación. En algunos ejemplos, el elemento sintáctico comprende un indicador de un bit. En algunos ejemplos, el elemento sintáctico se señala cuando la unidad de codificación se codifica utilizando una modalidad distinta a la modalidad de modulación de código de intrapulsos (IPCM). En otros ejemplos, el elemento sintáctico se señala solo cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación. En algunos ejemplos, un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio.

15 **[0188]** En algunos ejemplos, el elemento sintáctico puede no estar señalado. Por ejemplo, el elemento sintáctico puede no estar señalado cuando la unidad de codificación está intracodificada y cuando una modalidad de predicción de luma y una modalidad de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes. En otro ejemplo, el elemento sintáctico puede no ser señalado cuando la unidad de codificación está codificada con una modalidad de paleta. En estos ejemplos, el elemento sintáctico puede no estar presente en un flujo de bits recibido que comprende los datos de vídeo, y la decodificación del elemento sintáctico puede comprender deducir el valor del elemento sintáctico.

20 **[0189]** En algunos ejemplos, el elemento sintáctico es un primer elemento sintáctico y los datos de vídeo son un primer conjunto de datos de vídeo. En tales ejemplos, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar un valor de un índice de paleta para un primer píxel en una segunda unidad de codificación de un segundo conjunto de datos de vídeo. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además los valores de los índices de paleta para uno o más píxeles siguientes en un orden de recorrido inmediatamente posterior al primer píxel. La determinación de los valores de los índices de paleta puede comprender determinar los valores de los índices de paleta hasta que un píxel tenga un índice de paleta con un valor diferente al valor del índice de paleta para el primer píxel. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede determinar además una serie de valores de índice de paleta determinados para los siguientes píxeles. Según el número de valores de índice de paleta, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede codificar el índice de paleta utilizando una primera modalidad o una segunda modalidad. Cuando el número de valores de índice de paleta es mayor que uno, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar un valor de un segundo elemento sintáctico para indicar que la primera modalidad se usó para codificar el índice de paleta y fijar un valor de un tercer elemento sintáctico igual al número de valores de índice de paleta. Cuando el número de valores de índice de paleta es menor o igual a uno, la unidad de codificación por entropía 56 del codificador de vídeo 20 puede fijar el valor del segundo elemento sintáctico para indicar que la segunda modalidad se usó para codificar el índice de paleta. En algunos ejemplos, la primera modalidad es una modalidad de racha, y la segunda modalidad es una modalidad de píxel.

25 **[0190]** En algunos ejemplos, la codificación del elemento sintáctico puede comprender que la unidad de codificación por entropía 56 del codificador de vídeo 20 codifique datos de un mensaje de información de mejora suplementaria (SEI) que indica si la unidad de codificación fue codificada utilizando la conversión de color-espacio.

30 **[0191]** La figura 9 es un diagrama de flujo que ilustra un procedimiento a modo de ejemplo para decodificar un bloque actual de datos de vídeo. El bloque actual puede comprender una CU actual o una parte de la CU actual. Aunque se ha descrito con respecto al decodificador de vídeo 30 (figuras 1 y 3), debería entenderse que pueden configurarse otros dispositivos para realizar un procedimiento similar al de la figura 9.

35 **[0192]** La unidad de decodificación por Entropía 70 del decodificador de vídeo 30 puede recibir un primer bloque codificado de datos de vídeo (196). El primer bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación con pérdidas y un primer proceso de transformación de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede recibir además un segundo bloque codificado de datos de vídeo (198). El segundo bloque codificado de datos de vídeo se codificó utilizando una modalidad de codificación sin pérdidas y un segundo proceso de transformación de color-espacio.

40 **[0193]** El decodificador de vídeo 30 puede predecir el bloque actual (200), por ejemplo, usando una modalidad de intrapredicción o interpredicción para calcular un bloque predicho para el bloque actual. El decodificador de vídeo 30 también puede recibir datos codificados por entropía para el bloque actual, tales como datos codificados por entropía para coeficientes de un bloque residual correspondiente al bloque actual (202). El decodificador de vídeo 30 puede decodificar por entropía los datos codificados por entropía para reproducir los coeficientes del bloque residual (204). Posteriormente, el decodificador de vídeo 30 puede explorar de forma inversa los coeficientes reproducidos (206), para crear un bloque de coeficientes de transformación cuantizados. El decodificador de vídeo 30 puede entonces

transformar inversamente y cuantizar inversamente los coeficientes (208). La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar un elemento sintáctico para el bloque actual (210).

[0194] La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar además un proceso de transformación inversa de color-espacio al primer bloque codificado de datos de vídeo (212). La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar además el mismo proceso de transformación inversa de color-espacio al segundo bloque codificado de datos de vídeo (214). El decodificador de vídeo 30 puede finalmente decodificar el bloque actual combinando el bloque predicho y el bloque residual (216). En el lado del decodificador, puede que no sea necesario realizar un proceso de transformación directa de color-espacio, independientemente de la modalidad de codificación. En algunos ejemplos, el proceso de transformación inversa de color-espacio puede ser fijo. En algunos ejemplos, los bloques de datos de vídeo pueden ser unidades de codificación.

[0195] En algunos ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede modificar adicionalmente uno o más componentes del primer bloque codificado de datos de vídeo con un desplazamiento a la derecha de N bits o un desplazamiento a la izquierda de N bits. En algunos ejemplos, N puede ser igual a 1. En algunos ejemplos, los uno o más componentes del primer bloque codificado de datos de vídeo pueden ser dos componentes de croma.

[0196] En algunos ejemplos, el primer proceso de transformación de color-espacio comprende la aplicación de una matriz YCoCg. En otros ejemplos, el segundo proceso de transformación de color-espacio comprende la aplicación de una matriz YCoCg-R. En cualquiera de los dos ejemplos, el proceso de transformación inversa de color-espacio puede comprender un proceso de transformación inversa de color-espacio sin pérdidas. En algunos ejemplos, el proceso de transformación inversa sin pérdidas de color-espacio comprende la aplicación de una matriz YCoCg-R.

[0197] En algunos ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar un elemento sintáctico de una unidad de codificación de datos de vídeo. El elemento sintáctico puede indicar si la unidad de codificación se codificó utilizando la conversión de color-espacio. En algunos ejemplos, el elemento sintáctico puede comprender un indicador de un bit. En algunos ejemplos, la unidad de codificación se codifica en una modalidad diferente a la modalidad de modulación de código de intrapulso (IPCM) y el elemento sintáctico se señala solo para unidades de codificación que usan una modalidad diferente a la modalidad de IPCM. En algunos ejemplos, el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación. En algunos ejemplos, un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio. La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar además si un valor del elemento sintáctico indica que la unidad de codificación se codificó utilizando la conversión de color-espacio. En respuesta a la determinación de que el elemento sintáctico indica que la unidad de codificación se codificó utilizando conversión de color-espacio, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede aplicar el proceso de transformación inversa de color-espacio.

[0198] En algunos ejemplos, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación. Por ejemplo, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación cuando la unidad de codificación está intracodificada y cuando una modalidad de predicción de luma y una modalidad de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes. En otro ejemplo, el elemento sintáctico puede indicar que la conversión de color-espacio no se usó para codificar la unidad de codificación cuando la unidad de codificación se codifica con una modalidad de paleta. En estos ejemplos, el flujo de bits recibido puede no incluir el elemento sintáctico.

[0199] En algunos ejemplos, el elemento sintáctico es un primer elemento sintáctico y los datos de vídeo son un primer conjunto de datos de vídeo. En tales ejemplos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar además un valor de un segundo elemento sintáctico de un segundo conjunto de datos de vídeo. El segundo elemento sintáctico puede indicar si se usó una modalidad de paleta para codificar el segundo conjunto de datos de vídeo. En respuesta al segundo elemento sintáctico que indica que la modalidad de paleta se utilizó para codificar los datos de vídeo, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar el valor de un tercer elemento sintáctico. El tercer elemento sintáctico puede indicar si se utilizó una primera modalidad o una segunda modalidad para decodificar un índice de paleta para un píxel en la unidad de codificación. Basándose en el valor determinado del segundo elemento sintáctico, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar el índice de paleta utilizando la primera modalidad o la segunda modalidad. Cuando se utiliza la primera modalidad, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar un valor del índice de paleta, determinar un valor de un cuarto elemento sintáctico que indica un número de píxeles en un orden de recorrido que siga inmediatamente al píxel que se está decodificando actualmente, y duplicar el resultado de la determinación del valor del índice de paleta para los siguientes N píxeles en el orden de recorrido, con N igual al valor del cuarto elemento sintáctico. En algunos ejemplos, la primera modalidad es una modalidad de racha. Cuando se utiliza la segunda modalidad, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede determinar el valor del índice de paleta y generar un valor de muestra de píxel para el píxel, donde el valor de muestra de píxel es igual al valor del índice de paleta. En algunos ejemplos, la segunda modalidad es una modalidad de píxel.

[0200] En algunos ejemplos, la decodificación del elemento sintáctico comprende la unidad de decodificación por entropía 70 del decodificador de vídeo 30, configurada para decodificar datos de un mensaje de información de mejora suplementaria (SEI) que indica si la unidad de codificación se codificó utilizando la conversión de color-espacio.

[0201] Ha de reconocerse que, según el ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, fundirse u omitirse por completo (por ejemplo, no todos los actos o sucesos descritos son necesarios para la puesta en práctica de las técnicas). Además, en ciertos ejemplos, los actos o sucesos pueden realizarse simultáneamente, por ejemplo, mediante el procesamiento de múltiples hilos, el procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente.

[0202] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de estos. Si se implementan en software, las funciones, como una o más instrucciones o código, pueden almacenarse en, y 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 de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) medios de almacenamiento tangibles legibles por ordenador que son no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que uno o más ordenadores o uno o más procesadores puedan acceder para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0203] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe apropiadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una sede de la Red, un servidor u otro origen remoto mediante un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o unas tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas están incluidas en la definición de medio. Sin embargo, debería entenderse que los medios de 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. El término disco, como se usa en el presente documento, incluye un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, de los cuales algunos discos habitualmente reproducen datos magnéticamente, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

[0204] 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 de compuertas programables in situ (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. En consecuencia, el término «procesador», como se usa en el presente documento, puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos dedicados de hardware y/o software, configurados para codificar y decodificar, o incorporados en un códec combinado. Asimismo, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.

[0205] Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). En la presente divulgación se describen diversos componentes, módulos o unidades para poner de manifiesto aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización por diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec, o proporcionarse por medio de un grupo de unidades de hardware interoperativas, incluidos uno o más procesadores, como se ha descrito anteriormente, junto con software y/o firmware adecuados.

[0206] Se han descrito varios ejemplos de la divulgación. El alcance de la invención se define por las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento para codificar datos de vídeo en un proceso de codificación, el procedimiento que comprende:
- 5 determinar una modalidad de codificación usada para codificar un bloque actual de datos de vídeo, en donde la modalidad de codificación es una entre una modalidad de codificación con pérdidas y una modalidad de codificación sin pérdidas;
- 10 determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo;
- 15 aplicar un proceso de transformación de color-espacio YCoCg-R a un bloque residual para el bloque actual, cuando se determina que la modalidad de codificación es la modalidad de codificación con pérdidas;
- 20 aplicar un proceso de transformación de color-espacio YCoCg-R al bloque residual cuando se determina que la modalidad de codificación es la modalidad de codificación sin pérdidas;
- 25 aplicar un proceso de transformación inversa de color-espacio YCoCg-R en un ciclo de decodificación del proceso de codificación, independientemente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas; y
- cuando se determina que la modalidad de codificación es la modalidad de codificación con pérdidas, modificar un componente de Co y un componente de Cg de la matriz YCoCg, con un desplazamiento a la izquierda de 1 bit, antes de aplicar el proceso de transformación inversa de color-espacio en el bucle de decodificación del proceso de codificación.
2. El procedimiento según la reivindicación 1, que comprende además aumentar en 1 la profundidad de bits de un componente Co y un componente Cg cuando se determina que la modalidad de codificación es la modalidad de codificación sin pérdidas.
3. El procedimiento según la reivindicación 1, que comprende además determinar el uso de un proceso de transformación de color-espacio para codificar los datos de vídeo, en el que la determinación de usar un proceso de transformación de color-espacio para codificar los datos de vídeo comprende:
- 35 establecer un valor de un elemento sintáctico que indica que una unidad de codificación que comprende el bloque actual de los datos de vídeo se codificó utilizando un proceso de transformación de color-espacio, en donde el elemento sintáctico comprende un indicador de un bit, y en donde un valor de 1 para el elemento sintáctico indica que se utiliza un proceso de transformación de color-espacio para codificar la unidad de codificación.
4. El procedimiento según la reivindicación 3, en el que la unidad de codificación no está codificada en la modalidad de modulación de código de intrapulso (IPCM), y en el que el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio cuando la unidad de codificación no está codificada utilizando la modalidad de la modulación de código de intrapulso (IPCM); y / o
- 45 en el que el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación.
5. El procedimiento según la reivindicación 1, en el que un elemento sintáctico que indica que una unidad de codificación, que comprende el bloque actual de los datos de vídeo, se codificó utilizando un proceso de transformación de color-espacio, no se señala cuando la unidad de codificación está intracodificada y cuando una modalidad de predicción de luma y una modalidad de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes; y / o
- 55 en el que un elemento sintáctico, que indica que una unidad de codificación que comprende el bloque actual de los datos de vídeo se codificó utilizando un proceso de transformación del color-espacio, no se señala cuando la unidad de codificación se codifica con una modalidad de paleta.
6. Un dispositivo de codificación de vídeo que comprende:
- 60 una memoria configurada para almacenar datos de vídeo; y
- 65 uno o más procesadores configurados para:

determinar una modalidad de codificación usada para codificar un bloque actual de datos de vídeo, en donde la modalidad de codificación es uno entre una modalidad de codificación con pérdidas o una modalidad de codificación sin pérdidas;

5 determinar un proceso de transformación de color-espacio que depende de la modalidad de codificación utilizada para codificar los datos de vídeo;

10 aplicar un proceso de transformación de color-espacio YCoCg al bloque residual de datos para el bloque actual cuando se determina que la modalidad de codificación es la modalidad de codificación con pérdidas;

aplicar un proceso de transformación de color-espacio YCoCg-R al bloque residual cuando se determina que la modalidad de codificación es la modalidad de codificación sin pérdidas;

15 aplicar un proceso de transformación inversa de color-espacio YCoCg-R en un ciclo de decodificación del proceso de codificación, independientemente de si la modalidad de codificación es la modalidad de codificación con pérdidas o la modalidad de codificación sin pérdidas; y

20 cuando se determina que la modalidad de codificación es la modalidad de codificación con pérdidas, modificar un componente de Co y un componente de Cg de la matriz YCoCg, con un desplazamiento a la izquierda de 1 bit, antes de aplicar el proceso de transformación inversa de color-espacio en el bucle de decodificación del proceso de codificación.

25 7. El dispositivo de codificación de vídeo de la reivindicación 6, en el que los uno o más procesadores están configurados además para aumentar una profundidad de bit de un componente Co y un componente Cg en 1 cuando se determina que la modalidad de codificación es la modalidad de codificación sin pérdidas.

8. Un procedimiento de decodificación de datos de vídeo, el procedimiento que comprende:

30 recibir un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado es un bloque residual, y en donde el primer bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación con pérdidas y un proceso de transformación de color-espacio YCoCg;

35 recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado es un bloque residual, y en donde el segundo bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación sin pérdidas y un proceso de transformación de color-espacio YCoCg-R;

40 aplicar un proceso de transformación inversa de color-espacio YCoCg-R al primer bloque codificado de datos de vídeo; y

aplicar un proceso de transformación inversa de color-espacio YCoCg-R al segundo bloque codificado de datos de vídeo,

45 en donde el componente Co y el componente Cg del primer bloque codificado de datos de vídeo se modifican con un desplazamiento a la izquierda de 1 bit antes de aplicar el proceso de transformación inversa de color-espacio.

9. El procedimiento de la reivindicación 8, que comprende además:

50 decodificar un elemento sintáctico de una unidad de codificación de datos de vídeo, en donde la unidad de codificación comprende el primer bloque codificado o el segundo bloque codificado, y en donde el elemento sintáctico indica si la unidad de codificación fue codificada utilizando un proceso de transformación de color-espacio;

55 determinar si un valor del elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio; y

60 en respuesta a la determinación de que el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio, aplicar el proceso de transformación inversa de color-espacio, en donde el elemento sintáctico comprende un indicador de un bit,

y en donde un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio.

65 10. El procedimiento de la reivindicación 8, en el que una unidad de codificación que comprende el primer bloque codificado o el segundo bloque codificado no está codificada en la modalidad de modulación de código por

intrapulsos (IPCM), y en el que un elemento sintáctico que indica si la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio se señala solamente para unidades de codificación que no están codificadas utilizando la modalidad de IPCM; y / o en el que un elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio cuando hay coeficientes distintos de cero en una unidad de transformación de la unidad de codificación; y / o

en el que un elemento sintáctico indica que la unidad de codificación no se codificó utilizando un proceso de transformación del color-espacio cuando la unidad de codificación está intracodificada y cuando un modo de predicción de luma y un modo de predicción de croma de una unidad de predicción dentro de la unidad de codificación son diferentes, en donde el elemento sintáctico no está presente en un flujo de bits recibido que comprende los datos de vídeo, y en donde la decodificación del elemento sintáctico comprende inferir el valor del elemento sintáctico; y / o

en donde un elemento sintáctico indica que la unidad de codificación no fue codificada usando un proceso de transformación de color-espacio cuando la unidad de codificación está codificada con una modalidad de paleta, en donde el elemento sintáctico no está presente en un flujo de bits recibido que comprende los datos de vídeo, y en donde la decodificación del elemento sintáctico comprende deducir el valor del elemento sintáctico.

11. Un dispositivo de decodificación de vídeo que comprende:

una memoria configurada para almacenar datos de vídeo; y

uno o más procesadores configurados para:

recibir un primer bloque codificado de datos de vídeo, en donde el primer bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación con pérdidas y un proceso de transformación de color-espacio YCoCg;

recibir un segundo bloque codificado de datos de vídeo, en donde el segundo bloque codificado de datos de vídeo fue codificado usando una modalidad de codificación sin pérdidas y un proceso de transformación de color-espacio YCoCg-R;

aplicar un proceso de transformación inversa de color-espacio YCoCg-R al primer bloque codificado de datos de vídeo; y

aplicar un proceso de transformación inversa de color-espacio YCoCg-R al segundo bloque codificado de datos de vídeo,

en donde el componente Co y el componente Cg del primer bloque codificado de datos de vídeo se modifican con un desplazamiento a la izquierda de 1 bit antes de aplicar el proceso de transformación inversa de color-espacio.

12. El dispositivo de decodificación de vídeo de la reivindicación 11, en el que los uno o más procesadores están configurados además para:

decodificar un elemento sintáctico de una unidad de codificación de datos de vídeo, en donde la unidad de codificación comprende el primer bloque codificado o el segundo bloque codificado, en donde el elemento sintáctico indica si la unidad de codificación fue codificada utilizando un proceso de transformación de color-espacio;

determinar si un valor del elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio; y

en respuesta a la determinación de que el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio, aplicar el proceso de transformación inversa de color-espacio, en donde el elemento sintáctico comprende un indicador de un bit,

y en donde un valor de 1 para el elemento sintáctico indica que la unidad de codificación se codificó utilizando un proceso de transformación de color-espacio.

13. Un medio de almacenamiento legible por ordenador que comprende instrucciones que, cuando se ejecutan, hacen que uno o más procesadores de un dispositivo de decodificación de vídeo realicen el procedimiento de cualquiera de las reivindicaciones 8 a 10, o hacen que uno o más procesadores de un dispositivo de codificación de vídeo realicen el procedimiento de cualquiera de las reivindicaciones 1 a 5.

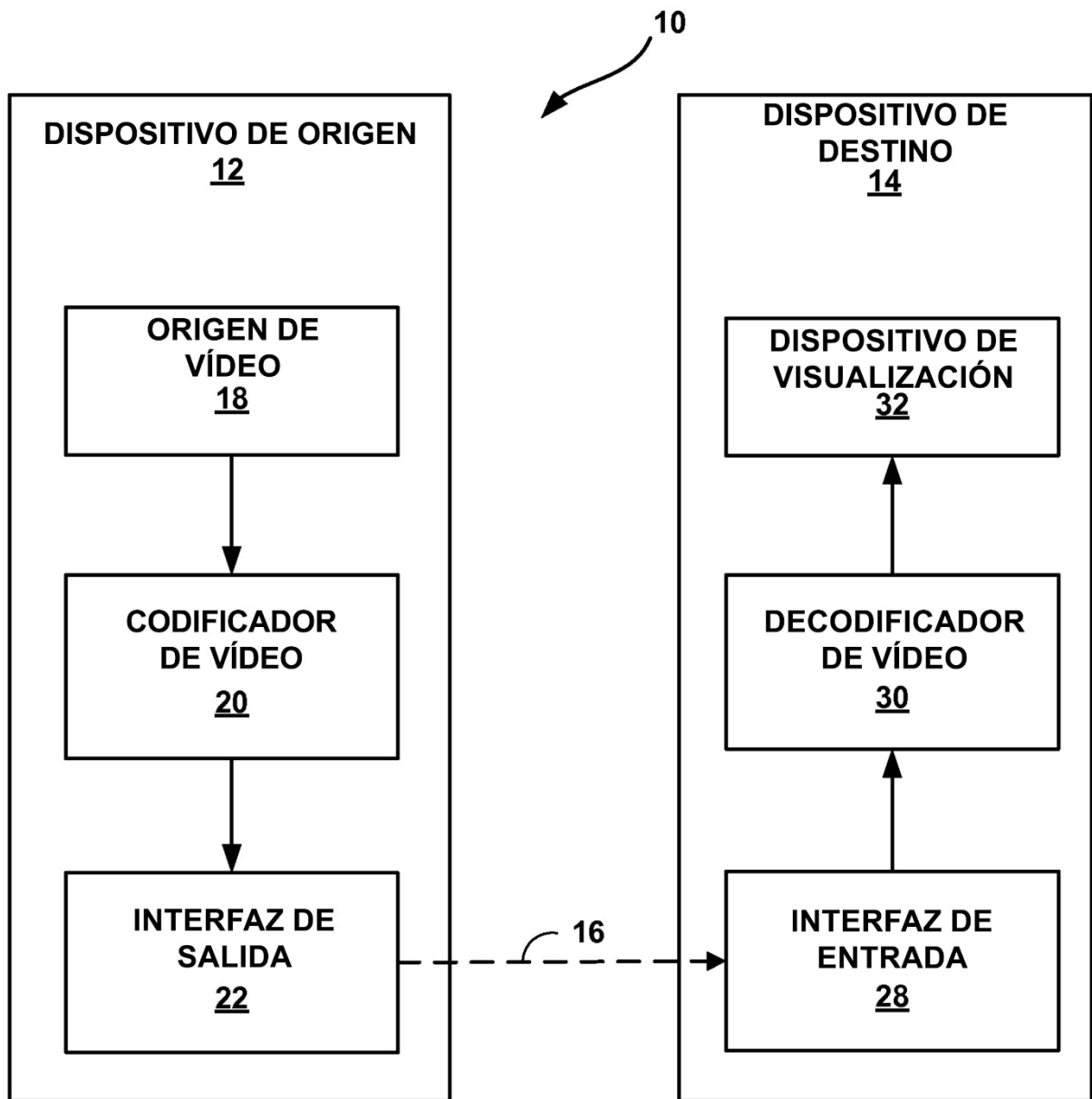


FIG. 1

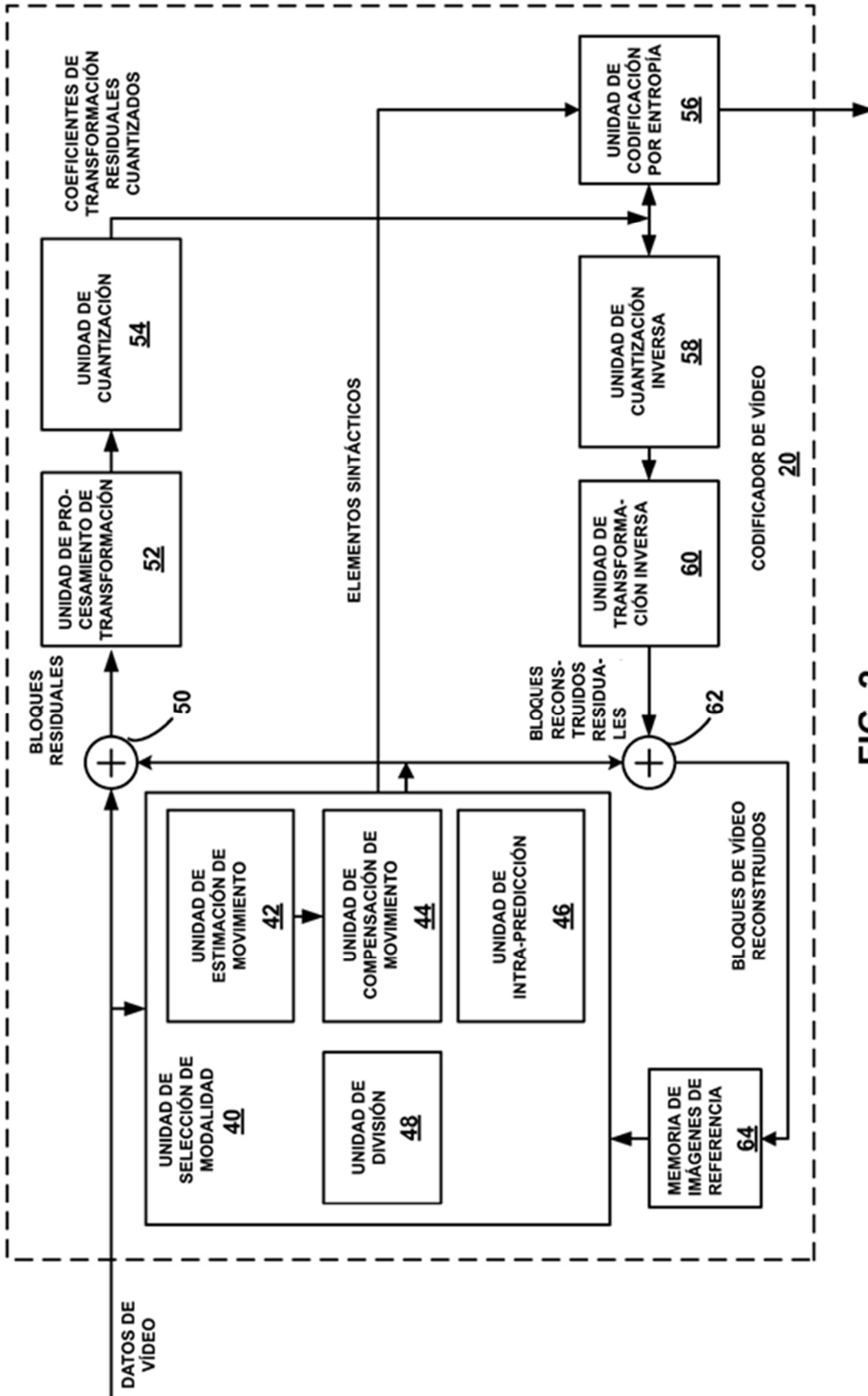


FIG. 2

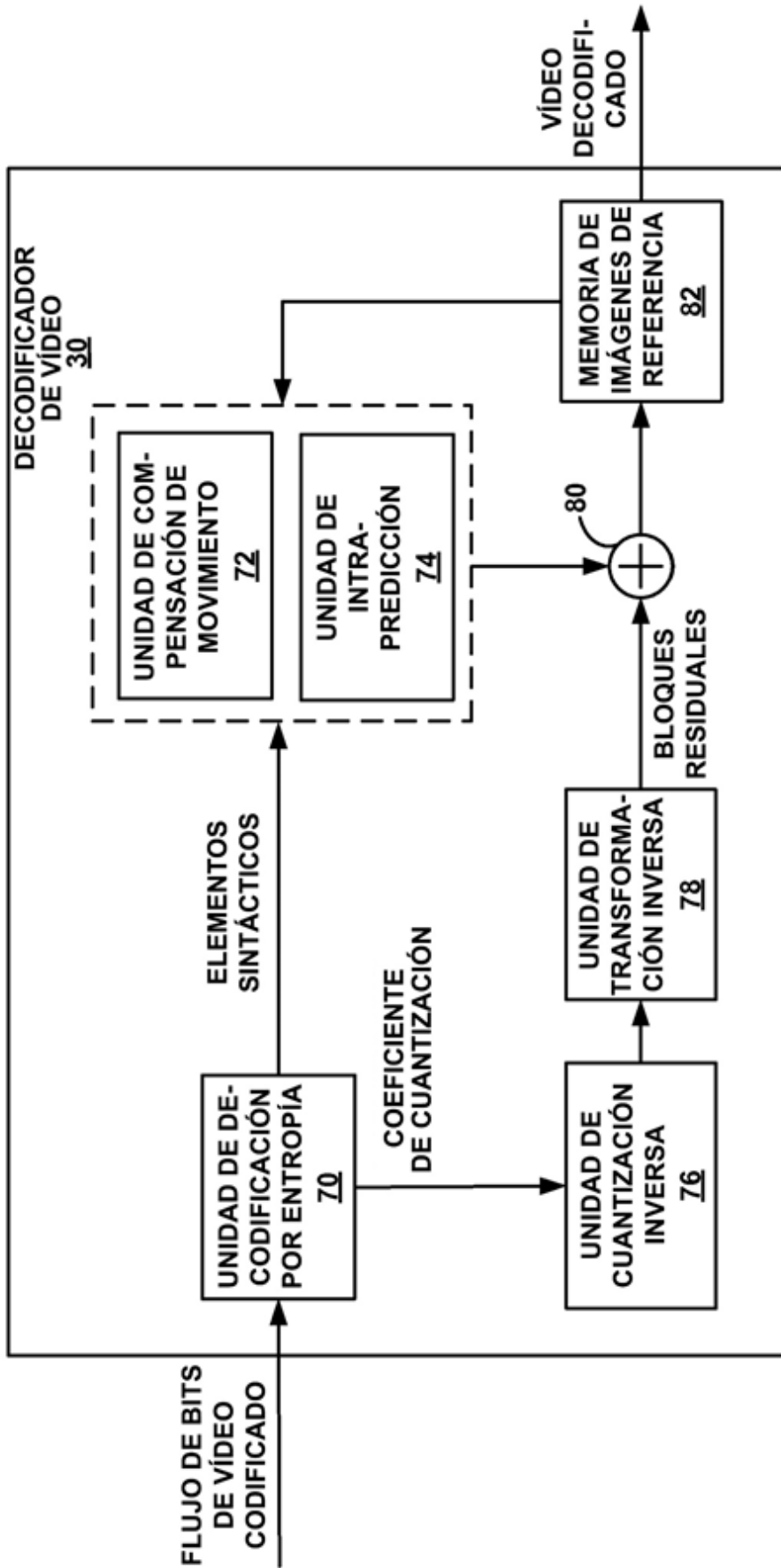


FIG. 3

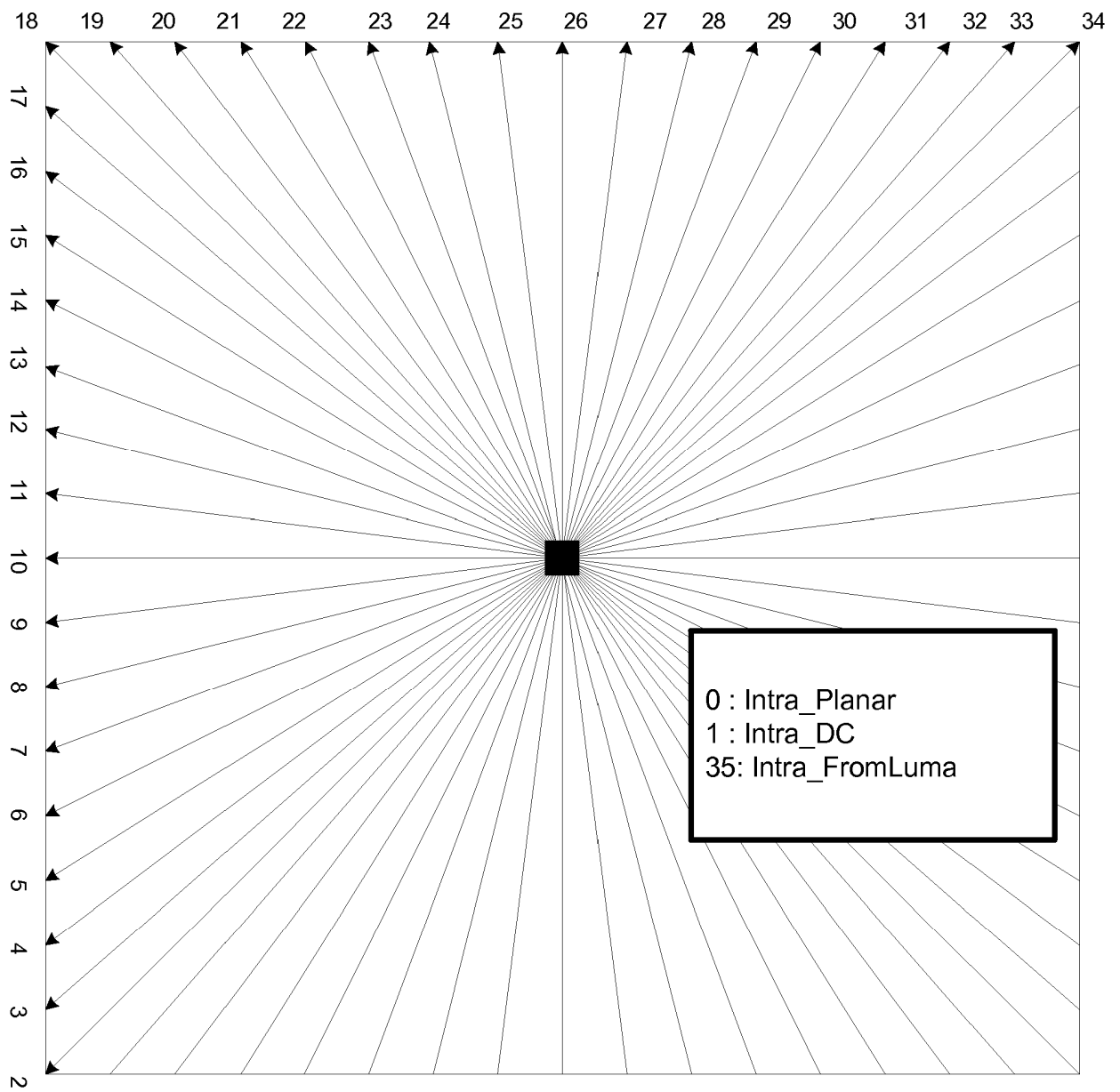


FIG. 4

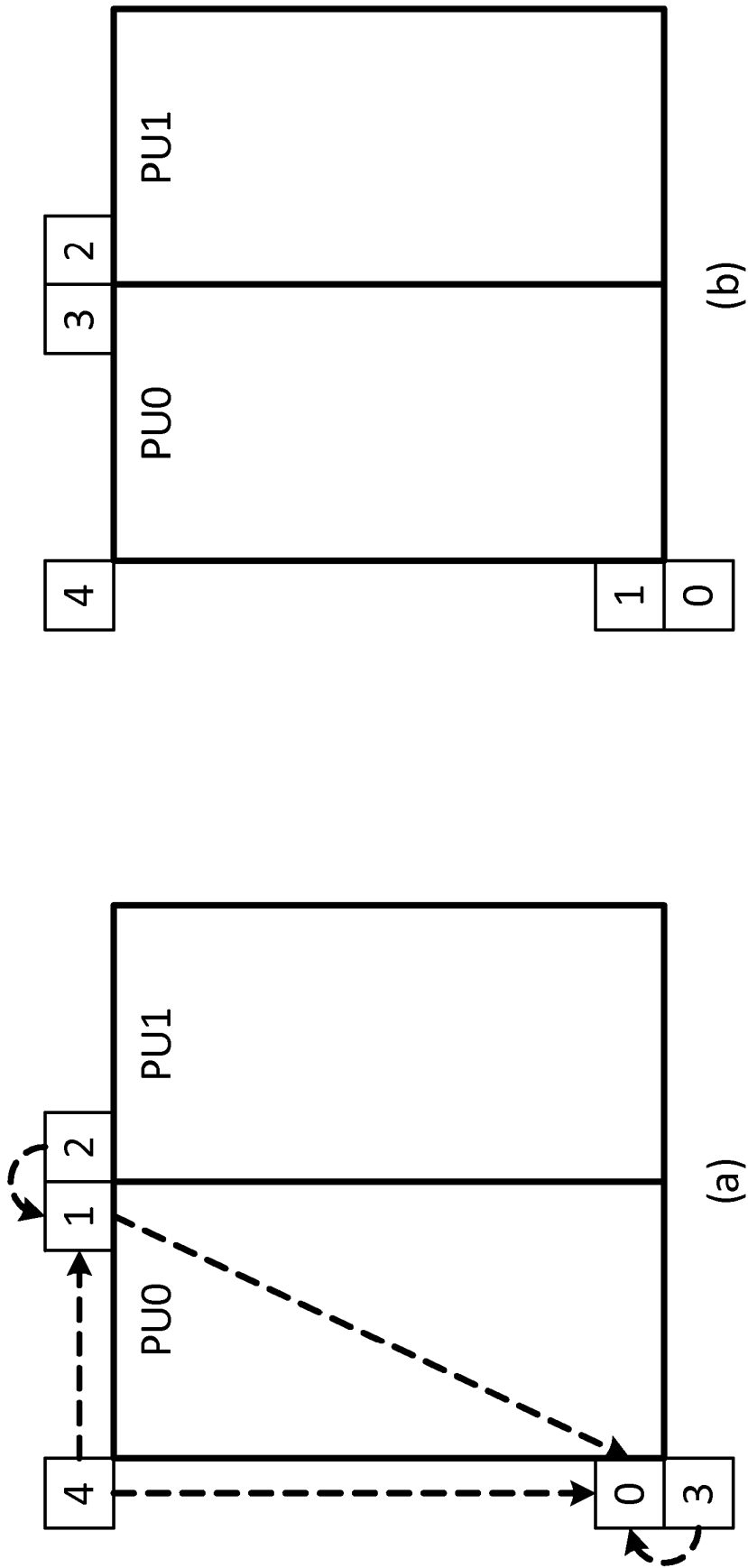


FIG. 5

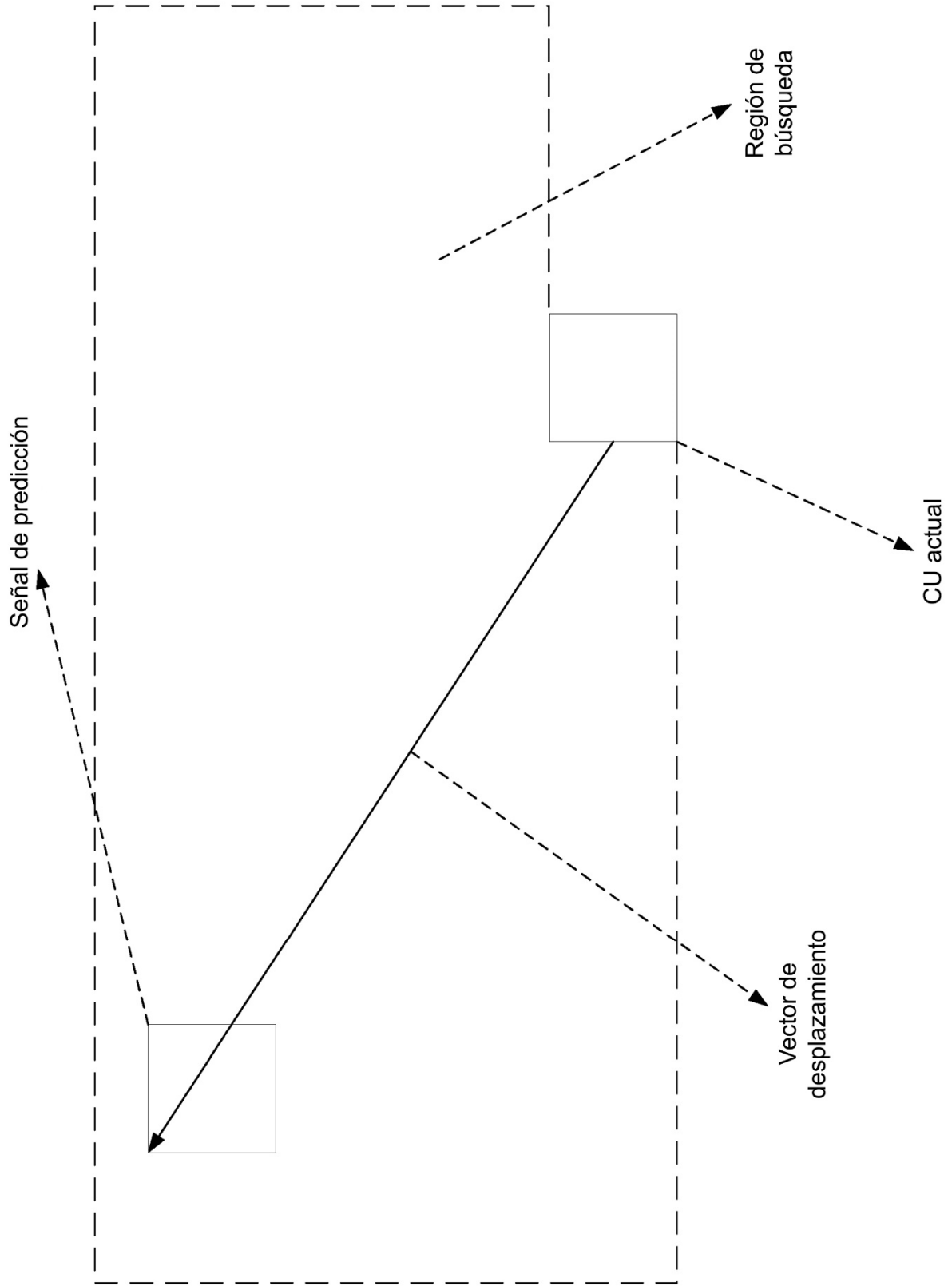


FIG. 6

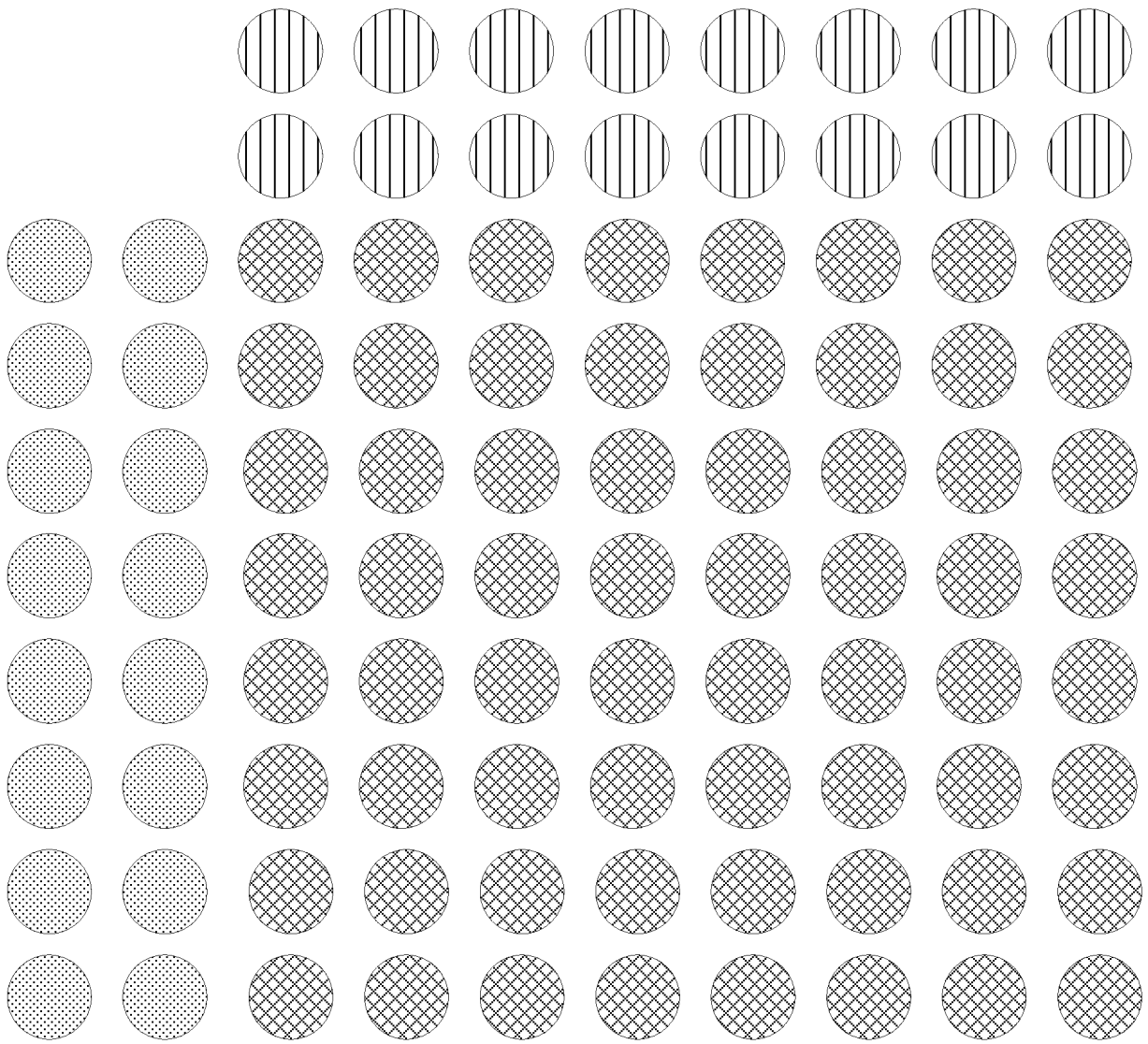


FIG. 7

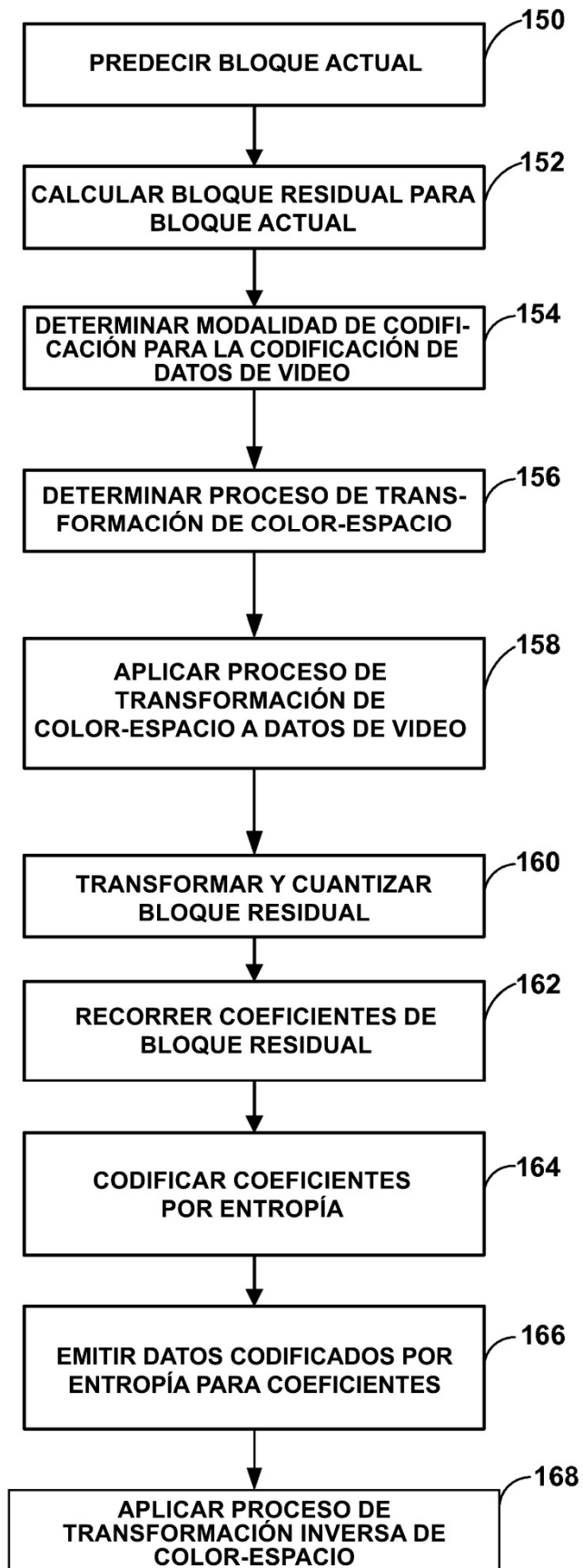


FIG. 8

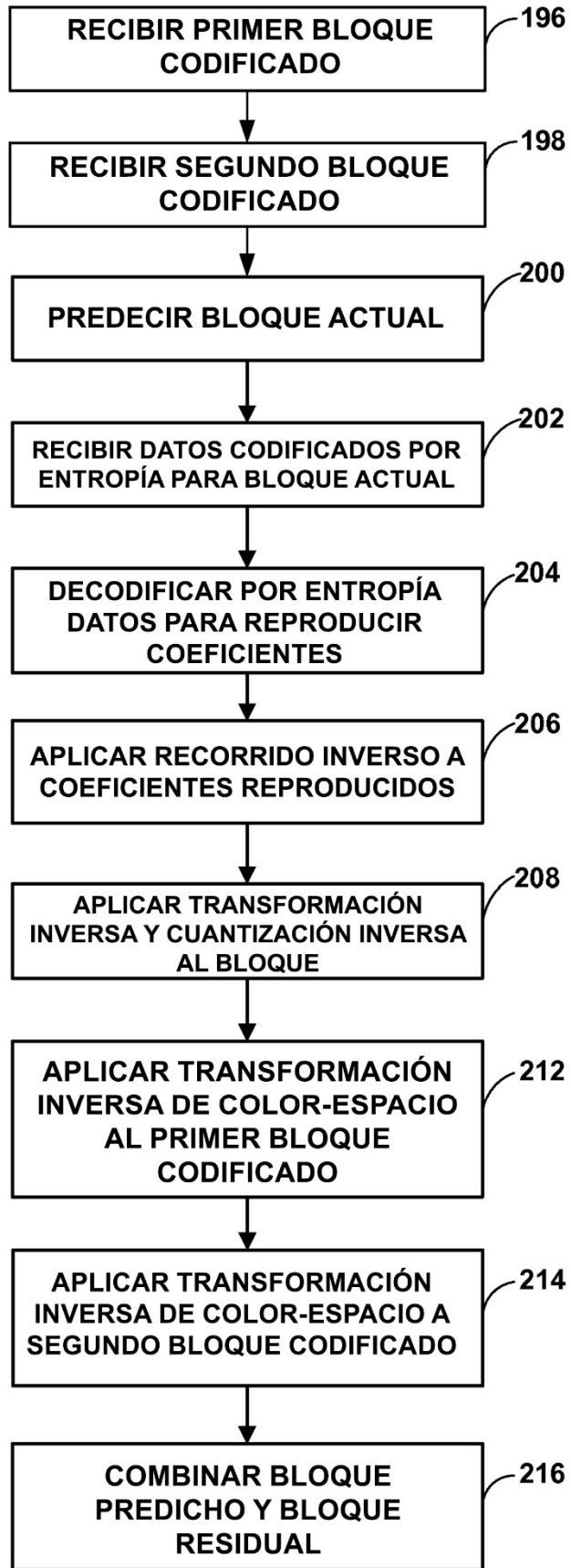


FIG. 9