

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 703 343**

51 Int. Cl.:

H04N 19/30 (2014.01)

H04N 19/186 (2014.01)

H04N 19/42 (2014.01)

H04N 19/70 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **17.12.2014 PCT/US2014/070910**

87 Fecha y número de publicación internacional: **25.06.2015 WO15095361**

96 Fecha de presentación y número de la solicitud europea: **17.12.2014 E 14825038 (4)**

97 Fecha y número de publicación de la concesión europea: **03.10.2018 EP 3085092**

54 Título: **Información de partición de señalización para la tabla de búsqueda 3D para la escalabilidad de gama de colores en la codificación de vídeo multicapas**

30 Prioridad:

17.12.2013 US 201361917228 P

30.05.2014 US 201462005845 P

16.12.2014 US 201414571939

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

08.03.2019

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)

5775 Morehouse Drive

San Diego, CA 92121-1714, US

72 Inventor/es:

LI, XIANG;

CHEN, JIANLE y

KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 703 343 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

Información de partición de señalización para la tabla de búsqueda 3D para la escalabilidad de gama de colores en la codificación de vídeo multicapas

SOLICITUDES RELACIONADAS

[0001] Esta solicitud reivindica el beneficio de la Solicitud Provisional de EE. UU N.º 61/917,228, presentada el 17 de diciembre de 2013, y la Solicitud Provisional de EE. UU N.º 62/005,845, presentada el 30 de mayo de 2014.

CAMPO TÉCNICO

[0002] Esta divulgación se refiere a la codificación de vídeo.

ANTECEDENTES

[0003] Las capacidades de vídeo digital pueden incorporarse a una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión digital directa, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos móviles o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión 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 de Vídeo Avanzada (AVC), la norma de Codificación de Vídeo de Alta Eficiencia (HEVC) y las extensiones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de codificación de vídeo.

[0004] Las técnicas de codificación de vídeo incluyen la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) 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 porción de una trama de vídeo) puede particionarse en bloques de vídeo, que también pueden denominarse bloques de árbol, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques adyacentes 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 adyacentes 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.

[0005] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se vaya a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se vaya a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y con los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel en un dominio de transformada, dando como resultado unos coeficientes de transformada residuales, que a continuación se pueden cuantizar. Los coeficientes de transformada cuantizados, dispuestos inicialmente en una matriz bidimensional, pueden escanearse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse codificación por entropía para lograr aún más compresión.

RESUMEN

[0006] En general, esta divulgación describe las técnicas para señalar información usada para generar tablas de consulta de color tridimensionales (3D) para la escalabilidad de gama de colores en la codificación de vídeo multicapas. Los codificadores de vídeo y/o los decodificadores de vídeo pueden usar técnicas de predicción de color para la escalabilidad de gama de colores para generar imágenes de referencia intercapas cuando una gama de colores para una capa inferior de datos de vídeo sea diferente de una gama de colores para una capa superior de datos de vídeo. Por ejemplo, un codificador de vídeo y/o un decodificador de vídeo pueden realizar primero la predicción de color usando una tabla de búsqueda 3D para la escalabilidad de gama de colores para convertir los datos de color de una imagen de referencia para la capa inferior en la gama de colores para la capa superior, y entonces generar imágenes de referencia intercapas basándose en los datos de color convertidos. De acuerdo con las técnicas descritas en esta divulgación, un codificador de vídeo puede codificar información de partición y/o valores de color de una tabla de búsqueda 3D generada para la escalabilidad de gama de colores. Un decodificador de vídeo puede decodificar la información de partición y/o los valores de color para generar la tabla de búsqueda 3D con el fin de realizar la escalabilidad de gama de colores.

[0007] En un ejemplo, esta divulgación está dirigida a un procedimiento de decodificación de datos de vídeo, comprendiendo el procedimiento determinar un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores; determinar un valor de partición de luma para un componente de luma de la tabla de búsqueda 3D; y generar la tabla de búsqueda 3D con una partición más gruesa para los componentes de croma y una partición más fina para el componente de luma, incluida la partición de cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base y en la partición de cada uno de los primeros números de octantes del componente de luma en un segundo número de octantes basándose en el valor de partición de luma. El procedimiento comprende además decodificar datos residuales de bloques de vídeo de los datos de vídeo; y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

[0008] En otro ejemplo, esta divulgación está dirigida a un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento generar una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores con la partición gruesa para los componentes de croma y con la partición más fina para un componente de luma, incluyendo la partición de cada componente de luma, un primer componente de croma y un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en un valor de partición base para la tabla de búsqueda 3D, y particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en un valor de partición de luma para el componente de luma de la tabla de búsqueda 3D. El procedimiento comprende además la predicción de bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D; y la codificación de datos residuales de los bloques de vídeo en un flujo de bits.

[0009] En un ejemplo adicional, esta divulgación está dirigida a un dispositivo de decodificación de vídeo que comprende una memoria configurada para almacenar datos de vídeo; y uno o más procesadores en comunicación con la memoria. El uno o más procesadores están configurados para determinar un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo, determinar un valor de partición de luma para un componente de luma de la tabla de búsqueda 3D y generar el tabla de búsqueda 3D con la partición más gruesa para los componentes de croma y con una partición más fina para el componente de luma, el uno o más procesadores configurados para particionar cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base, y particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en el valor de partición de luma. El uno o más procesadores están configurados además para decodificar datos residuales de bloques de vídeo de los datos de vídeo, y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

[0010] En otro ejemplo, esta divulgación está dirigida a un dispositivo de codificación de vídeo que comprende una memoria configurada para almacenar datos de vídeo; y uno o más procesadores en comunicación con la memoria. Los uno o más procesadores están configurados para generar una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo con particiones más gruesas para los componentes de croma y con particiones más finas para un componente de luma, el uno o más procesadores configurados para particionar cada uno del componente de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en un valor de partición base, y particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en un valor de partición de luma para el componente de luma de la tabla de búsqueda 3D. El uno o más procesadores están configurados para predecir bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D, y codificar datos residuales de los bloques de vídeo en un flujo de bits.

[0011] En un ejemplo adicional, la presente divulgación se refiere a un dispositivo de decodificación de vídeo que comprende medios para determinar un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores; medios para determinar un valor de partición de luma para un componente de luma de la tabla de búsqueda 3D; y medios para generar la tabla de búsqueda 3D con una partición más gruesa para los componentes de croma y con una partición más fina para el componente de luma, incluidos los medios para particionar cada uno de los componentes de luma, de un primer componente de croma y un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base, y medios para particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en el valor de partición de luma. El dispositivo de decodificación de vídeo comprende además medios para decodificar datos residuales de bloques de vídeo de los datos de vídeo; y medios para reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

[0012] En un ejemplo adicional, esta divulgación está dirigida a un medio de almacenamiento legible por ordenador que almacena instrucciones para los datos de vídeo de decodificación que, al ejecutarse, causan que uno o más procesadores determinen un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores; determinen un valor de partición de luma para un componente de luma de la tabla

de búsqueda 3D; y generen la tabla de búsqueda 3D con particiones más gruesas para los componentes de croma y con una partición más fina para el componente de luma, las instrucciones causan que uno o más procesadores particionen cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base, y
 5 particionen cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en el valor de partición de luma. Las instrucciones además causan que uno o más procesadores decodifiquen datos residuales de bloques de vídeo de los datos de vídeo; y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y en al menos una imagen de referencia generada usando la tabla de búsqueda 3D.

10 **[0013]** En otro ejemplo, esta divulgación se dirige hacia un procedimiento de decodificación de datos de vídeo, comprendiendo el procedimiento determinar un número de octantes para cada uno de tres componentes de color de una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores; para cada uno de los octantes para cada uno de los componentes de color, decodificar los coeficientes de asignación de color para una función de
 15 asignación de color lineal de los valores de color en la tabla de búsqueda 3D usada para convertir datos de color en una primera gama de colores para una capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo; y generar la tabla de búsqueda 3D basándose en el número de octantes para cada uno de los componentes de color y en los valores de color asociados con los coeficientes de asignación de color para cada uno de los octantes. El procedimiento comprende además decodificar datos residuales de bloques de vídeo de los datos de vídeo; y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

20 **[0014]** En un ejemplo adicional, la presente divulgación se refiere a un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento la generación de una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores basándose un número de octantes para cada uno de tres componentes de color y valores de color para cada uno de los octantes; y para cada uno de los octantes para cada uno de los componentes de color, codificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D usada para convertir datos de color en una primera gama de colores para una
 25 capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo. El procedimiento comprende además predecir bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D; y codificar datos residuales de los bloques de vídeo en un flujo de bits.

30 **[0015]** En un ejemplo adicional, esta divulgación está dirigida a un dispositivo de decodificación de vídeo que comprende una memoria configurada para almacenar datos de vídeo; y uno o más procesadores en comunicación con la memoria. El uno o más procesadores están configurados para determinar un número de octantes para cada uno de los tres componentes de color de una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo, para cada uno de los octantes para cada uno de los componentes de color, decodificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la
 35 tabla de búsqueda 3D usada para convertir datos de color en una primera gama de colores para una capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo, y generar la tabla de búsqueda 3D basándose en el número de octantes para cada uno de los componentes de color y en los valores de color asociados con los coeficientes de asignación de color para cada uno de los octantes. El uno o más procesadores están configurados además para decodificar datos residuales de bloques de vídeo de los datos de vídeo, y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

40 **[0016]** En un ejemplo adicional, la presente divulgación está dirigida a un dispositivo de codificación de vídeo que comprende una memoria configurada para almacenar datos de vídeo; y uno o más procesadores en comunicación con la memoria. El uno o más procesadores están configurados para generar una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo basándose en una cantidad de octantes para cada uno de los tres componentes de color y en los valores de color para cada uno de los octantes; y para cada uno de los octantes para cada uno de los componentes de color, codificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D usada para convertir datos de
 45 color en una primera gama de colores para una capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo. El uno o más procesadores están configurados para predecir bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D, y codificar datos residuales de los bloques de vídeo en un flujo de bits.

50 **[0017]** En otro ejemplo, esta divulgación está dirigida a un dispositivo de decodificación de vídeo que comprende medios para determinar un número de octantes para cada uno de tres componentes de color de una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores; medios para decodificar, para cada uno de los octantes para cada uno de los componentes de color, los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D usada para convertir datos de color en una primera
 55 gama de colores para una capa inferior del vídeo datos en una segunda gama de colores para una capa superior de los datos de vídeo; y medios para generar la tabla de búsqueda 3D basándose en el número de octantes para cada

uno de los componentes de color y en los valores de color asociados con los coeficientes de asignación de color para cada uno de los octantes. El dispositivo de decodificación de vídeo comprende además medios para decodificar datos residuales de bloques de vídeo de los datos de vídeo; y medios para reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

[0018] En un ejemplo adicional, la presente divulgación está dirigida a un medio de almacenamiento legible por ordenador que almacena instrucciones para los datos de vídeo de decodificación que, al ejecutarse, causan que uno o más procesadores determinen un número de octantes para cada uno de tres componentes de color de una tabla de consulta tridimensional (3D) para la escalabilidad de gama de colores; para cada uno de los octantes para cada uno de los componentes de color, que decodifique los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D usada para convertir datos de color en una primera gama de colores para una capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo; y que genere la tabla de búsqueda 3D basándose en el número de octantes para cada uno de los componentes de color y en los valores de color asociados con los coeficientes de asignación de color para cada uno de los octantes. Las instrucciones además causan que uno o más procesadores decodifiquen datos residuales de bloques de vídeo de los datos de vídeo; y reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos en una imagen de referencia generada usando la tabla de búsqueda 3D.

[0019] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetivos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0020]

La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y de decodificación de vídeo de ejemplo que puede utilizar técnicas para la tabla de consulta 3D basándose en la escalabilidad de gama de colores.

La FIG. 2 es una ilustración conceptual que muestra un ejemplo de escalabilidad en tres dimensiones diferentes.

La FIG. 3 es una ilustración conceptual que muestra una estructura de ejemplo de un flujo de bits de codificación de vídeo escalable.

La FIG. 4 es una ilustración conceptual que muestra unidades de acceso de codificación de vídeo escalables de ejemplo en orden de flujo de bits.

La FIG. 5 es un diagrama de bloques que ilustra una extensión de codificación de vídeo escalable de ejemplo para el codificador HEVC (SHVC).

La FIG. 6 es un gráfico que ilustra una gama de colores de ejemplo de una secuencia de vídeo de muestra.

La FIG. 7 es un diagrama de bloques que ilustra la conversión de la gama de colores de alta definición (HD) BT.709 en la gama de colores de alta definición (UHD) BT.2020.

La FIG. 8 es un diagrama de bloques que ilustra un codificador escalable de gama de colores que incluye una unidad de procesamiento de predicción de color que puede generar una imagen de referencia intercapas cuando una gama de colores de capa base y una gama de colores de capa de mejora son diferentes.

Las FIGS. 9(a) y 9(b) son ilustraciones conceptuales que muestran una tabla de consulta 3D de ejemplo para la escalabilidad de gama de colores.

La FIG. 10 es una ilustración conceptual que muestra la interpolación trilineal con una tabla de búsqueda 3D para la escalabilidad de gama de colores.

La FIG. 11 es una ilustración conceptual que muestra la interpolación tetraédrica con una tabla de búsqueda 3D para la escalabilidad de gama de colores.

La FIG. 12 es una ilustración conceptual que muestra seis ejemplos de tetraedros usados para abarcar un punto P de una tabla de búsqueda 3D que se vaya a interpolar usando interpolación tetraédrica.

La FIG. 13 es una ilustración conceptual que muestra una tabla de búsqueda 3D de ejemplo con un componente de luma particionado independientemente y componentes de croma particionados conjuntamente.

La FIG. 14 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas para usar la escalabilidad de gama de colores basándose en tablas de búsqueda 3D en la codificación de vídeo multicapas.

5 La FIG. 15 es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas para usar la tabla de búsqueda 3D basada en la escalabilidad de gama de colores en la codificación de vídeo multicapa.

10 La FIG. 16 es un diagrama de flujo que ilustra una operación de ejemplo de codificación de información de partición para al menos uno de los componentes de color de una tabla de búsqueda 3D.

La FIG. 17 es un diagrama de flujo que ilustra una operación de ejemplo de decodificación de información de partición para al menos uno de los componentes de color de una tabla de búsqueda 3D.

15 La FIG. 18 es un diagrama de flujo que ilustra una operación de ejemplo de codificación de valores de color para cada uno de los octantes para cada uno de los componentes de color de una tabla de búsqueda 3D.

20 La FIG. 19 es un diagrama de flujo que ilustra una operación de ejemplo de decodificación de valores de color para cada uno de los octantes para cada uno de los componentes de color de una tabla de búsqueda 3D.

DESCRIPCIÓN DETALLADA

25 **[0021]** Esta divulgación describe técnicas para predicción de color tridimensional (3D) para la escalabilidad de gama de colores en la codificación de vídeo de multicapas. La codificación de vídeo multicapas puede estar de acuerdo con la norma de codificación de vídeo de alta eficiencia (HEVC), incluyendo cualquier extensión de codificación de vídeo escalable, una extensión de codificación de vídeo multivistas, una extensión de codificación de vídeo 3D (es decir, codificación de vídeo multivistas más profundidad) u otras extensiones de codificación de vídeo multicapas para HEVC. Los codificadores de vídeo y/o decodificadores de vídeo pueden usar las técnicas para generar imágenes de referencia intercapas cuando una gama de colores para una capa inferior de datos de vídeo sea diferente de una gama de colores para una capa superior de datos de vídeo. En algunos ejemplos, las técnicas también se pueden usar cuando la profundidad de bits de la capa inferior de datos de vídeo sea diferente de la profundidad de bits de la capa superior de los datos de vídeo.

35 **[0022]** Una gama de colores comprende una gama completa de colores que se pueden reproducir para una imagen, por ejemplo, en una imagen, un fragmento, un bloque o una capa de datos de vídeo. Convencionalmente, en la codificación de vídeo multicapas, una capa inferior de datos de vídeo (por ejemplo, una capa base) y una capa superior de los datos de vídeo (por ejemplo, una capa de mejora) incluyen datos de color en la misma gama de colores, por ejemplo, gama de colores BT.709 de alta definición (HD). En este caso, un codificador de vídeo y/o un decodificador de vídeo pueden generar imágenes de referencia intercapas para la capa superior de los datos de vídeo como versiones de muestra ascendente de imágenes de referencia cúbicadas para la capa inferior de los datos de vídeo.

40 **[0023]** En algunos ejemplos, sin embargo, una capa inferior de los datos de vídeo puede incluir datos de color en una primera gama de colores, por ejemplo, BT.709, y una capa superior de los datos de vídeo pueden incluir datos de color en una segunda gama de colores diferente, por ejemplo, una gama de colores de definición ultra alta (UHD) BT.2020. En este ejemplo, para generar imágenes de referencia intercapas para la capa superior de los datos de vídeo, un codificador de vídeo y/o un decodificador de vídeo primero debe realizar una predicción de color para convertir los datos de color de una imagen de referencia en la primera gama de colores para la capa inferior de los datos de vídeo a la segunda gama de colores para la capa superior de los datos de vídeo.

50 **[0024]** El codificador de vídeo y/o el decodificador de vídeo puede realizar la predicción de color usando una tabla de búsqueda 3D para la escalabilidad de gama de colores. En algunos ejemplos, se puede generar una tabla de búsqueda 3D independiente para cada uno de los componentes de color, es decir, un componente de luma (Y), un primer componente de croma (U) y un segundo componente de croma (V). Cada una de las tablas de búsqueda 3D incluye una dimensión de luma (Y), una primera dimensión de croma (U) y una segunda dimensión de croma (V), y se indexa usando los tres componentes de color independientes (Y, U, V).

55 **[0025]** Convencionalmente, las tablas de búsqueda 3D son siempre simétricas de manera que las tablas de búsqueda 3D tienen un mismo tamaño para el componente de luma, el primer componente de croma y el segundo componente de croma. Además, de manera convencional, las tablas de búsqueda 3D siempre están equilibrada, de manera que un tamaño de cada dimensión de las tablas de búsqueda 3D es siempre el mismo. Esto puede dar como resultado tamaños de tablas grandes con alta complejidad computacional y altos costes de señalización. Por ejemplo, los tamaños de tablas pueden ser de hasta 9x9x9 o 17x17x17.

60 **[0026]** En la Solicitud de Patente de EE. UU N.º 14/512,177 (Despacho de Abogado No. 1212-712US01/140193), presentada el 10 de octubre de 2014, se describen técnicas que permiten que un codificador de vídeo y/o un decodificador de vídeo generen una tabla de búsqueda 3D asimétrica y/o desequilibrada de manera que la tabla de

búsqueda 3D tenga un tamaño que sea diferente para el componente de luma que para el primer componente de croma y el segundo componente de croma. El codificador de vídeo y/o el decodificador de vídeo pueden generar esta tabla de búsqueda 3D asimétrica y/o no balanceada al particionar el componente de luma en un número diferente de segmentos que los primer y segundo componentes de croma. En este ejemplo, los tamaños de tabla pueden ser de hasta 8x2x2.

[0027] Las técnicas de la presente divulgación se dirigen hacia la señalización de la información usada para generar tablas de búsqueda 3D para la escalabilidad de gama de colores. De acuerdo con las técnicas, un codificador de vídeo puede codificar información de partición y/o valores de color de una tabla de búsqueda 3D generada para la escalabilidad de gama de colores. Un decodificador de vídeo puede decodificar la información de partición y/o los valores de color para generar la tabla de búsqueda 3D con el fin de realizar la escalabilidad de gama de colores. Las técnicas descritas en esta divulgación pueden ser particularmente útiles para señalar la información usada para generar tablas de búsqueda 3D asimétricas y/o desequilibradas.

[0028] En un ejemplo de las técnicas divulgadas, un decodificador de vídeo y/o un codificador de vídeo pueden generar una tabla de búsqueda 3D con la partición gruesa para los primer y segundo componentes de croma y con la partición más fina para un componente de luma mediante la partición de cada uno de los componentes de color en varios octantes de acuerdo con un valor de partición base, por ejemplo, una profundidad de división máxima para la tabla de búsqueda 3D, y entonces una partición adicional de cada uno de los octantes del componente de luma basándose en un valor de partición luma. De esta manera, los componentes de croma de la tabla de búsqueda 3D se particionan en un número menor o en menos octantes (es decir, partición más gruesa) y el componente de luma de la tabla de búsqueda 3D se particiona en un número mayor o en más octantes (es decir, partición más fina).

[0029] En un ejemplo, el valor de partición de luma puede señalizarse en un flujo de bits por el codificador de vídeo al decodificador de vídeo. En otros ejemplos, el valor de partición base también puede señalizarse en el flujo de bits por el codificador de vídeo al decodificador de vídeo. En otros casos, el valor de partición de luma se puede derivar tanto del codificador de vídeo como del decodificador de vídeo y/o el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo como en el decodificador de vídeo.

[0030] Como ejemplo, el valor de partición base es igual a 1 de manera que cada uno de los componentes de color de primer croma, de segundo croma y de luma se particionan en un único octante, y el valor de partición de luma es igual a 4 de manera que el único octante del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 4x1x1. Como otro ejemplo, el valor de partición base es igual a 2, de manera que cada uno de los componentes de color de primer croma, de segundo croma y de luma se particiona en dos octantes, y el valor de partición de luma es igual a 4, de manera que cada uno de los dos octantes del componente de luma se divide en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 8x2x2. Como puede verse, un valor de partición inferior da como resultado una partición más gruesa (es decir, un número menor de octantes) para un componente de color.

[0031] De acuerdo con las técnicas, cada uno de los componentes de color puede particionarse en uno o más octantes basándose en uno o más del valor de partición base o del valor de partición de luma. En esta divulgación, el término "octante" se define como una región tridimensional que incluye ocho vértices. En esta divulgación, los términos "partición", "octante", "segmento" y "cuboide" se pueden usar indistintamente para describir las regiones particionadas de los componentes de color de la tabla de búsqueda 3D.

[0032] Además, basándose en al menos uno de los primer o segundo componentes de croma de la tabla de búsqueda 3D que se particiona en más de un octante, es decir, el valor de partición base es mayor que uno, el codificador de vídeo puede indicar un límite de partición para el de los componentes de croma al decodificador de vídeo. El límite de partición define una partición desigual de uno de los componentes de croma en dos o más octantes. En otras palabras, uno o ambos componentes de croma no se pueden dividir en dos o más octantes iguales o de igual tamaño. En este caso, para un componente dado de uno de los componentes de croma, al menos uno de los octantes tiene un tamaño diferente al de uno o más octetos. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo sólo señala el límite de partición basándose en la condición de que uno de los componentes de croma se divide en más de un octante. De lo contrario, el límite de partición es innecesario y no se señala al decodificador de vídeo.

[0033] En otro ejemplo de las técnicas divulgadas, un codificador de vídeo y/o un decodificador de vídeo puede generar una tabla de búsqueda 3D basándose en una serie de octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma, y en los valores de color para cada uno de los octantes. Como se describió anteriormente, en algunos casos, el codificador de vídeo también puede señalar el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D al decodificador de vídeo. Para que el decodificador de vídeo determine los valores de color en la tabla de búsqueda 3D, los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D se señalan por el codificador de vídeo al decodificador de vídeo. La función de asignación de color lineal se usa para convertir datos de color en una primera gama de colores para una capa inferior de datos de vídeo en una segunda gama de colores para una capa superior de datos de vídeo, y los coeficientes de asignación de color son factores de ponderación entre los componentes de color de la parte inferior y capas superiores de los datos de vídeo. Para cada uno de los componentes

de color, uno de los coeficientes de asignación de color puede ser un coeficiente clave que defina un factor de ponderación entre el mismo componente de color de las capas inferior y más alta de los datos de vídeo.

5 **[0034]** Los coeficientes de asignación de color de la función de asignación de color lineal se derivan como valores de punto flotante. Antes de señalar los coeficientes de asignación de color en un flujo de bits, los valores de punto flotante se pueden convertir en valores enteros. Aunque los valores enteros pueden ser menos precisos que los valores de punto flotante, los valores enteros son más fáciles de señalar y las operaciones de números enteros son menos costosas computacionalmente que las operaciones de punto flotante. La conversión puede usar una profundidad de bits para los valores enteros basándose en al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D. Además, los valores de los coeficientes de asignación de color pueden restringirse para que estén dentro de un rango dado basándose en un valor fijo predefinido o en un valor dependiente de al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.

15 **[0035]** Uno o más de los coeficientes de asignación de color se puede predecir de manera que los valores residuales entre los valores originales de los coeficientes de asignación de color y los valores del color de coeficientes de asignación se codifican en el flujo de bits previsto. Por ejemplo, para un primer octante para cada uno de los componentes de color, los coeficientes de asignación de color de la función de asignación de color lineal pueden predecirse basándose en valores fijos predefinidos. En un ejemplo, un coeficiente clave de la función de asignación de color lineal puede predecirse basándose en un valor predicho igual a un valor distinto de cero predefinido, y cualquier coeficiente de asignación de color restante puede predecirse basándose en un valor predicho igual a cero. En este ejemplo, los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color se pueden predecir basándose en los valores predichos de al menos un octante anterior, tal como el primer octante. En algunos casos, los valores residuales de los coeficientes de asignación de color pueden cuantizarse basándose en un valor de cuantización determinado. El codificador de vídeo puede señalar el valor de cuantización determinado para que el decodificador de vídeo realice una cuantización inversa para decodificar adecuadamente los coeficientes de asignación de color.

30 **[0036]** Entre las normas de codificación de vídeo se incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluyendo sus extensiones de codificación de vídeo ajustable a escala (SVC) y de codificación de vídeo multivistas (MVC).

35 **[0037]** Ha finalizado el diseño de una nueva norma de codificación de vídeo, concretamente la HEVC, por parte el Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC. Un borrador de la norma HEVC denominado Borrador de trabajo 10 (WD10) de la HEVC, de Bross et al., "High Efficiency Video Coding (HEVC) Text Specification Draft 10 (for FDIS & Last Call)" [Memoria descriptiva textual de la Codificación de Vídeo de Alta Eficacia (HEVC), Borrador 10]", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 12ª conferencia: Ginebra, CH, 14-23 de enero de 2013, JCTVC-L1003v34, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip. La norma HEVC finalizada se conoce como versión 1 de HEVC.

45 **[0038]** Un informe por defecto, Wang et al., "High efficiency video coding (HEVC) Defect Report [Informe por Defecto de Codificación de Vídeo de Alta Eficiencia (HEVC)]" Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 14ª reunión: Viena, AT, 25 de julio-2 de agosto de 2013, JCTVC-N1003v1, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zip. El documento estándar finalizado de HEVC se publica como ITU-T H.265, Serie H: Sistemas audiovisuales y multimedia, Infraestructura de servicios audiovisuales - Codificación de vídeo en movimiento, Codificación de vídeo de alta eficiencia, Sector de normalización de las telecomunicaciones de la Unión Internacional de Telecomunicaciones (ITU), abril de 2013.

55 **[0039]** La extensión multivistas a HEVC (MV-HEVC) y otra extensión de HEVC para la codificación de vídeo 3D más avanzada (3D-HEVC) se están desarrollando por el JCT-3V. Una memoria descriptiva de borrador de MV-HEVC, denominado Borrador de Trabajo 5 (WD5) de MV-HEVC, Tech et al., "MV-HEVC Draft Text 5 [Texto de borrador 5 de MV-HEVC]", Equipo de colaboración conjunta sobre desarrollo de extensión de codificación de vídeo 3D (JCT-3V) del ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 5ª reunión: Viena, AT, 27 de julio-2 de agosto de 2013, JCT3V-E1004v6, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/5_Vienna/wg11/JCT3V-E1004-v6.zip. Un borrador de especificación de 3D-HEVC, denominado Borrador de Trabajo 1 (WD1) de 3D-HEVC y descrito en Tech et al., "3D-HEVC Draft Text 1 [Texto de borrador de D-HEVC 1]" Equipo de colaboración conjunta sobre desarrollo de extensión de codificación de vídeo 3D (JCT-3V) del ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 5ª. Reunión: Viena, AT, 27 de julio-2 de agosto de 2013, JCT3V-E1001v3, está disponible en http://phenix.it-sudparis.eu/jct2/doc_end_user/documents/5_Vienna/wg11/JCT3V-E1001-v3.zip.

65 **[0040]** La extensión escalable a HEVC (SHVC) también se está desarrollando por el JCT-VC. Una memoria descriptiva de borrador de la SHVC, denominada Borrador de Trabajo 3 (WD3) de la SHVC, Chen et al., "SHVC Draft 3 [Borrador

3 de la SHVC]", Equipo de colaboración conjunta sobre codificación de vídeo (JCT-VC) del ITU-T SG16 WP3 e ISO/CEI JTC1/SC29/WG11, 14ª Reunión: Viena, AT, 25 de julio-2 de agosto de 2013, JCTVC-N1008v3, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1008-v3.zip.

5 **[0041]** La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y de decodificación de vídeo 10 de ejemplo que puede utilizar técnicas para la escalabilidad de gama de colores basada en tablas de búsqueda 3D. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados que vayan a codificarse 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 de entre una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, teléfonos tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de transmisión en continuo de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

10 **[0042]** El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se vayan 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 habilite 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, inalámbrica o cableada, 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 enrutadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

20 **[0043]** 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 del dispositivo de almacenamiento mediante una interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera de 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 del dispositivo de almacenamiento mediante transmisión continua o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Entre los ejemplos de servidores de archivos se incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectado en red (NAS) o una unidad de disco local. El dispositivo 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 WiFi), una conexión alámbrica (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

30 **[0044]** 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 de entre una diversidad de aplicaciones multimedia, tales como radiodifusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo por 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 soportar la transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como la transmisión en continuo de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

40 **[0045]** En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. 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 estar configurado para aplicar las técnicas para procesamiento de datos de vídeo en paralelo. 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 una fuente de vídeo externa 18, tal como una cámara externa. Del mismo modo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0046] El sistema ilustrado 10 de la FIG. 1 es meramente un ejemplo. Las técnicas para procesamiento de datos de vídeo en paralelo pueden realizarse mediante cualquier dispositivo de codificación y/o de decodificación de vídeo digital. Aunque, en general, las técnicas de esta divulgación se realizan mediante un dispositivo de codificación de vídeo, las técnicas también pueden realizarse mediante un codificador/decodificador de vídeo, denominado típicamente «CÓDEC». Además, las técnicas de esta divulgación también pueden realizarse 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 incluye componentes de codificación y de decodificación de vídeo. Por lo tanto, el sistema 10 puede soportar la transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo para transmisión de vídeo en continuo, reproducción de vídeo, radiodifusión de vídeo o videotelefonía.

[0047] La fuente 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 contenga vídeo grabado previamente y/o una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenidos de vídeo. Como alternativa adicional, la fuente 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 la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, 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 alámbricas. En cada caso, el codificador de vídeo 20 puede codificar el vídeo captado, precaptado o generado por ordenador. La información de vídeo codificada puede emitirse entonces mediante la interfaz de salida 22 hacia un medio legible por ordenador 16.

[0048] El medio legible por ordenador 16 puede incluir medios transitorios, tales como una difusión inalámbrica o una transmisión de 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 forma 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 contenga 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 diversos ejemplos.

[0049] 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 se usa por 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 visualiza los datos de vídeo decodificados a un usuario, y puede comprender cualquiera de una variedad de 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.

[0050] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse cada uno como cualquiera de una variedad de circuitería de codificadores adecuada, tal como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno de 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.

[0051] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 funcionan de acuerdo con una norma de compresión de vídeo, tal como la ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluida su extensión de codificación de vídeo escalable (SVC), su extensión de codificación de vídeo multivistas (MVC) y su extensión de vídeo tridimensional (3DV) basada en MVC. En algunos casos, cualquier flujo de bits que se ajuste a 3DV basada en MVC siempre contiene un subflujo de bits que es compatible con un perfil MVC, por ejemplo, un alto perfil estéreo. Además, se están emprendiendo unas iniciativas para generar una extensión de codificación 3DV para H.264/AVC, en concreto, 3DV basada en AVC. Entre otras normas de codificación de vídeo de ejemplo se incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264, ISO/IEC Visual.

[0052] En el ejemplo de la FIG. 1, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con la norma de Codificación de Vídeo de Alta Eficiencia (HEVC) elaborada por el Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y del Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC. La memoria descriptiva del borrador de HEVC, mencionada anteriormente, se denomina Borrador de Trabajo 10 (WD10) de HEVC, y la versión finalizada de la norma de HEVC se denomina versión 1 de HEVC. El JCT-3V está desarrollando MV-HEVC y 3D-HEVC. Una memoria descriptiva de borrador reciente de MV-HEVC se denomina MV-HEVC WD5, y una memoria descriptiva de borrador reciente de 3D-HEVC se denomina 3D-HEVC WD1. La SHVC se está desarrollando por el JCT-VC. Una memoria descriptiva de borrador reciente de SHVC se conoce como SHVC WD3.

[0053] En HEVC y en otras normas de codificación de vídeo, una secuencia de vídeo incluye típicamente una serie de imágenes. Las imágenes también pueden denominarse "tramas". Una imagen puede incluir tres matrices de muestras, denominadas S_L , S_{Cb} y S_{Cr} . S_L es una matriz bidimensional (es decir, un bloque) de muestras de luma. S_{Cb} es una matriz bidimensional de muestras de crominancia Cb. S_{Cr} es una matriz bidimensional de muestras de crominancia Cr. Las muestras de crominancia también se pueden mencionar en el presente documento como muestras de "croma". En otros casos, una imagen puede ser monocromática y puede incluir solo una matriz de muestras de luminancia.

[0054] El codificador de vídeo 20 puede generar un conjunto de unidades de árbol de codificación (CTU). Cada una de las CTU puede comprender un bloque de árbol de codificación de muestras de luma, dos bloques de árbol de codificación correspondientes de muestras de croma y estructuras sintácticas usadas para codificar las muestras de los bloques de árbol de codificación. En una imagen monocromática o en una imagen que tenga tres planos de color separados, una CTU puede comprender un único bloque de árbol de codificación y estructuras sintácticas usadas para codificar las muestras del bloque de árbol de codificación. Un bloque de árbol de codificación puede ser un bloque de muestras de tamaño $N \times N$. Una CTU también puede denominarse "bloque de árbol" o "máxima unidad de codificación" (LCU). Las CTU de la HEVC pueden ser análogas ampliamente a los macrobloques de otras normas de codificación de vídeo, tales como H.264/AVC. Sin embargo, una CTU no está limitada necesariamente a un tamaño particular y puede incluir una o más unidades de codificación (CU). Un fragmento puede incluir un número entero de CTU ordenadas consecutivamente en el escaneo de cuadrículado.

[0055] Esta divulgación puede usar el término "unidad de vídeo", "bloque de vídeo" o "bloque" para referirse a uno o más bloques de muestras y estructuras sintácticas usadas para codificar muestras del uno o más bloques de muestras. Los tipos de ejemplo de unidades de vídeo pueden incluir CTU, CU, PU, unidades de transformada (TU) en HEVC, O macrobloques, particiones de macrobloque, etc.

[0056] Para generar una CTU codificada, el codificador de vídeo 20 puede realizar de forma recursiva una partición de árbol cuaternario en los bloques de árbol de codificación de una CTU para dividir los bloques de árbol de codificación en bloques de codificación, de ahí el nombre "unidades de árbol de codificación." Un bloque de codificación es un bloque de muestras de tamaño $N \times N$. Una CU puede comprender un bloque de codificación de muestras de luma y dos bloques de codificación correspondientes de muestras de croma de una imagen que tenga una matriz de muestras de luma, una matriz de muestras de Cb y una matriz de muestras de Cr y estructuras sintácticas usadas para codificar las muestras de los bloques de codificación. En imágenes monocromáticas o imágenes que tengan tres planos de color independientes, una CU puede comprender un único bloque de codificación y estructuras sintácticas usadas para codificar las muestras del bloque de codificación.

[0057] El codificador de vídeo 20 puede particionar un bloque de codificación de una CU en uno o más bloques predictivos. Un bloque predictivo puede ser un bloque rectangular (es decir, cuadrado o no cuadrado) de muestras en el que se aplique la misma predicción. Una unidad de predicción (PU) de una CU puede comprender un bloque predictivo de muestras de luma, dos bloques predictivos correspondientes de muestras de croma de una imagen y estructuras sintácticas usadas para predecir las muestras de bloques predictivos. En imágenes monocromáticas o imágenes que tengan tres planos de color independientes, una PU puede comprender un único bloque predictivo y estructuras sintácticas usadas para predecir el bloque predictivo. El codificador de vídeo 20 puede generar bloques predictivos de luma, Cb y Cr para bloques predictivos de luma, Cb y Cr de cada PU de la CU.

[0058] El codificador de vídeo 20 puede usar intrapredicción o interpredicción para generar los bloques predictivos para una PU. Si el codificador de vídeo 20 usa la intrapredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras decodificadas de la imagen asociada con la PU.

[0059] Si el codificador de vídeo 20 usa interpredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras decodificadas de una o más imágenes distintas a la imagen asociada con la PU. La interpredicción puede ser una interpredicción unidireccional (es decir, unipredicción) o una interpredicción bidireccional (es decir, bipredicción). Para realizar la unipredicción o la bipredicción, el codificador de vídeo 20 puede generar una primera lista de imágenes de referencia (RefPicList0) y una segunda lista de imágenes de referencia (RefPicList1) para un fragmento actual.

[0060] Cada una de las listas de imágenes de referencia pueden incluir una o más imágenes de referencia. Cuando se usa la unipredicción, el codificador de vídeo 20 puede buscar las imágenes de referencia en RefPicList0 o RefPicList1 o en ambas para determinar una ubicación de referencia dentro de una imagen de referencia. Además, cuando se usa la unipredicción, el codificador de vídeo 20 puede generar, basándose al menos en parte en las muestras correspondientes a la ubicación de referencia, los bloques predictivos de muestras para la PU. Además, cuando se usa la unipredicción, el codificador de vídeo 20 puede generar un único vector de movimiento que indique un desplazamiento espacial entre un bloque predictivo de la PU y la ubicación de referencia. Para indicar el desplazamiento espacial entre un bloque predictivo de la PU y la ubicación de referencia, un vector de movimiento puede incluir un componente horizontal que especifique un desplazamiento horizontal entre el bloque predictivo de la PU y la ubicación de referencia y puede incluir un componente vertical que especifique un desplazamiento vertical entre el bloque predictivo de la PU y la ubicación de referencia.

[0061] Cuando se usa la bipredicción para codificar una PU, el codificador de vídeo 20 puede determinar una primera ubicación de referencia en una imagen de referencia en RefPicList0 y una segunda posición de referencia en una imagen de referencia en RefPicList1. El codificador de vídeo 20 puede generar entonces, basándose al menos en parte en muestras correspondientes a las primera y segunda ubicaciones de referencia, los bloques predictivos para la PU. Además, cuando se usa la bipredicción para codificar la PU, el codificador de vídeo 20 puede generar un primer movimiento que indique un desplazamiento espacial entre un bloque de muestra de la PU y la primera ubicación de referencia y un segundo movimiento que indique un desplazamiento espacial entre el bloque predictivo de la PU y la segunda ubicación de referencia.

[0062] Después de que el codificador de vídeo 20 genera bloques predictivos de luma, de Cb y de Cr para una o más PU de una CU, el codificador de vídeo 20 puede generar un bloque residual de luma para la CU. Cada muestra en el bloque residual de luma de la CU indica una diferencia entre una muestra de luma en uno de los bloques predictivos de luma de la CU y una muestra correspondiente en el bloque de codificación de luma original de la CU. Además, el codificador de vídeo 20 puede generar un bloque residual de Cb para la CU. Cada muestra en el bloque residual de Cb de la CU puede indicar una diferencia entre una muestra de Cb en uno de los bloques predictivos de Cb de la CU y una muestra correspondiente en el bloque de codificación de Cb original de la CU. El codificador de vídeo 20 también puede generar un bloque residual de Cr para la CU. Cada muestra en el bloque residual de Cr de la CU puede indicar una diferencia entre una muestra de Cr en uno de los bloques predictivos de Cr de la CU y una muestra correspondiente en el bloque de codificación de Cr original de la CU.

[0063] Además, el codificador de vídeo 20 puede usar la partición en árbol cuádruple para descomponer los bloques residuales de luma, de Cb y de Cr de una CU en uno o más bloques de transformada de luma, de Cb y de Cr. Un bloque de transformada puede ser un bloque rectangular de muestras en el que se aplique la misma transformada. Una unidad de transformada (TU) de una CU puede comprender un bloque de transformada de muestras de luma, dos bloques de transformada correspondientes de muestras de croma y estructuras sintácticas usadas para transformar las muestras de bloques de transformada. En imágenes monocromáticas o imágenes que tengan tres planos de color independientes, una PU puede comprender un único bloque predictivo y estructuras sintácticas usadas para predecir el bloque predictivo. De este modo, cada TU de una CU puede estar asociada con un bloque de transformada de luma, un bloque de transformada de Cb y un bloque de transformada de Cr. El bloque de transformada de luma asociado con la TU puede ser un subbloque del bloque residual de luma de la CU. El bloque de transformada de Cb puede ser un subbloque del bloque residual de Cb de la CU. El bloque de transformada de Cr puede ser un subbloque del bloque residual de Cr de la CU.

[0064] El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de luma de una TU para generar un bloque de coeficientes de luma para la TU. Un bloque de coeficientes puede ser una matriz bidimensional de coeficientes de transformada. Un coeficiente de transformada puede ser una cantidad escalar. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cb de una TU para generar un bloque de coeficientes de Cb para la TU. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cr de una TU para generar un bloque de coeficientes de Cr para la TU.

[0065] Después de generar un bloque de coeficientes (por ejemplo, un bloque de coeficientes de luma, un bloque de coeficientes de Cb o un bloque de coeficientes de Cr), el codificador de vídeo 20 puede cuantizar el bloque de coeficientes. La cuantización se refiere, en general, a un proceso en el que los coeficientes de transformada se cuantizan para reducir posiblemente la cantidad de datos usados para representar los coeficientes de transformada, proporcionando una compresión adicional. Además, el codificador de vídeo 20 puede cuantizar inversamente los coeficientes de transformada y aplicar una transformada inversa a los coeficientes de transformada para reconstruir bloques de transformada de TU de CU de una imagen. La unidad de reconstrucción 20 puede usar los valores residuales de los bloques de transformada de las TU de una CU y los bloques predictivos de las PU de la CU para reconstruir los bloques de codificación de la CU. Mediante la reconstrucción de los bloques de codificación para cada CU de una imagen, el decodificador de vídeo 20 puede reconstruir la imagen. El codificador de vídeo 20 puede almacenar imágenes reconstruidas en una memoria intermedia de imágenes decodificadas (DPB). El codificador de vídeo 20 puede usar imágenes reconstruidas en la DPB para la interpredicción y la intrapredicción.

[0066] Después de que el codificador de vídeo 20 cuantifique un bloque de coeficientes, el codificador de vídeo 20 puede codificar por entropía elementos sintácticos que indiquen los coeficientes de transformada cuantizados. Por ejemplo, el codificador de vídeo 20 puede realizar la Codificación Aritmética Binaria Adaptativa al Contexto (CABAC) en los elementos sintácticos que indiquen los coeficientes de transformada cuantizados. El codificador de vídeo 20 puede emitir los elementos sintácticos codificados por entropía en un flujo de bits.

[0067] El codificador de vídeo 20 puede emitir un flujo de bits que incluya una secuencia de bits que forme una representación de imágenes codificadas y datos asociados. El flujo de bits puede comprender una secuencia de unidades de capa de abstracción de red (NAL). Cada una de las unidades NAL incluye un encabezado de unidad NAL y encapsula una carga útil de secuencia de octetos sin procesar (RBSP). El encabezado de unidad NAL puede incluir un elemento sintáctico que indique un código de tipo de unidad NAL. El código de tipo de unidad NAL especificado por el encabezado de unidad NAL de una unidad NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contenga un número entero de bytes que se encapsule dentro de una unidad NAL. En algunos casos, una RBSP incluye cero bits.

[0068] Diferentes tipos de unidades NAL pueden encapsular diferentes tipos de RBSP. Por ejemplo, un primer tipo de unidad NAL puede encapsular una RBSP para un conjunto de parámetros de imagen (PPS), un segundo tipo de unidad NAL puede encapsular una RBSP para un fragmento codificado, un tercer tipo de unidad NAL puede encapsular una RBSP para información de mejora complementaria (SEI), y así sucesivamente. Un PPS es una estructura sintáctica que puede contener elementos sintácticos que se apliquen a cero o más secuencias de vídeo codificadas completas. Las unidades NAL que encapsulen las RBSP para datos de codificación de vídeo (a diferencia de las RBSP para conjuntos de parámetros y mensajes SEI) pueden denominarse unidades NAL de la capa de codificación de vídeo (VCL). Una unidad NAL que encapsula un fragmento codificado puede denominarse en el presente documento unidad NAL de fragmento codificado. Una RBSP para un fragmento codificado puede incluir un encabezado de fragmento y datos de fragmento.

[0069] El decodificador de vídeo 30 puede recibir un flujo de bits. Además, el decodificador de vídeo 30 puede analizar sintácticamente el flujo de bits para decodificar elementos sintácticos del flujo de bits. El decodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo basándose, al menos en parte, en los elementos sintácticos decodificados a partir del flujo de bits. El proceso para reconstruir los datos de vídeo puede ser, en general, recíproco al proceso realizado por el codificador de vídeo 20. Por ejemplo, el decodificador de vídeo 30 puede usar vectores en movimiento de las PU para determinar bloques predictivos para las PU de una CU actual. El decodificador de vídeo 30 puede usar un vector de movimiento o vectores de movimiento de las PU para generar bloques predictivos para las PU.

[0070] Además, el decodificador de vídeo 30 puede invertir bloques de coeficientes de cuantización asociados con TU de la CU actual. El decodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes de transformada para reconstruir los bloques de transformada asociados a las TU de la CU actual. El decodificador de vídeo 30 puede reconstruir los bloques de codificación de la CU actual añadiendo las muestras de los bloques predictivos de muestra para las PU de la CU actual a las muestras correspondientes de los bloques de transformada de las TU de la CU actual. Mediante la reconstrucción de los bloques de codificación para cada CU de una imagen, el decodificador de vídeo 30 puede reconstruir la imagen. El decodificador de vídeo 30 puede almacenar imágenes decodificadas en una memoria intermedia de imágenes decodificadas para su salida y/o para su uso en la decodificación de otras imágenes.

[0071] En MV-HEVC, 3D-HEVC y SHVC, un codificador de vídeo puede generar un flujo de bits de múltiples capas que comprenda una serie de unidades de la capa de abstracción de red (NAL). Las diferentes unidades NAL del flujo de bits pueden estar asociadas con diferentes capas del flujo de bits. Una capa se puede definir como un conjunto de unidades NAL de capa de codificación de vídeo (VCL) y unidades NAL sin VCL asociadas que tengan el mismo identificador de capa. Una capa puede ser equivalente a una vista en la codificación de vídeo multivistas. En la codificación de vídeo multivistas, una capa puede contener todos los componentes de vista de la misma capa con diferentes casos de tiempo. Cada componente de vista puede ser una imagen codificada de la escena de vídeo que pertenece a una vista específica en un caso de tiempo específico. En la codificación de vídeo 3D, una capa puede contener todas las imágenes de profundidad codificadas de una vista específica o imágenes de texturas codificadas de una vista específica. De forma similar, en el contexto de la codificación de vídeo escalable, una capa corresponde típicamente a imágenes codificadas que tengan características de vídeo diferentes a las imágenes codificadas en otras capas. Dichas características de vídeo típicamente incluyen la resolución espacial y el nivel de calidad (relación señal-ruido). En HEVC y en sus extensiones, la escalabilidad temporal se puede lograr dentro de una capa definiendo un grupo de imágenes con un nivel temporal particular como una subcapa.

[0072] Para cada una de las capas, los datos en una capa inferior pueden decodificarse sin referencia a los datos en cualquier capa superior. Los datos en una capa inferior pueden decodificarse sin referencia a los datos en cualquier capa superior en la codificación de vídeo escalable, por ejemplo. Las unidades NAL solo encapsulan datos de una única capa. En la SHVC, una vista puede denominarse "capa base" si un decodificador de vídeo puede decodificar imágenes en la vista sin referencia a imágenes en cualquier otro punto de vista. La capa base puede ajustarse a la memoria descriptiva básica de HEVC. Por tanto, las unidades NAL que encapsulan los datos de la capa restante más

alta del flujo de bits pueden eliminarse del flujo de bits sin afectar a la decodificabilidad de los datos en las capas restantes del flujo de bits. En MV-HEVC y 3D-HEVC, las capas superiores pueden incluir componentes de vista adicionales. En SHVC, las capas superiores pueden incluir datos de mejora de la relación señal/ruido (SNR), datos de mejora espacial y/o datos de mejora temporal.

5 **[0073]** En algunos ejemplos, los datos en una capa superior pueden decodificarse con referencia a los datos en una o más capas inferiores. Las capas inferiores se pueden usar como imágenes de referencia para comprimir la capa superior usando la predicción intercapas. Los datos de las capas inferiores se pueden muestrear para tener la misma resolución que las capas superiores. En general, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden realizar una predicción intercapas de manera similar a la predicción intercapas descrita anteriormente, excepto que una o más capas inferiores muestreadas hacia arriba pueden usarse como imágenes de referencia en lugar de una o más imágenes adyacentes.

15 **[0074]** FIG. 2 es una ilustración conceptual que muestra un ejemplo de escalabilidad en tres dimensiones diferentes. En una estructura de codificación de vídeo escalable, las escalabilidades se habilitan en tres dimensiones. En el ejemplo de la FIG. 2, las escalabilidades están habilitadas en una dimensión espacial (S) 100, en una dimensión temporal (T) 102 y en una relación señal/ruido (SNR) o dimensión de calidad (Q) 104. En la dimensión temporal 102, las velocidades de trama con 7,5 Hz (T0), 15 Hz (T1) o 30 Hz (T2), por ejemplo, pueden recibir soporte de la escalabilidad temporal. Cuando se soporta la escalabilidad espacial, diferentes resoluciones como QCIF (S0), CIF (S1) y 4CIF (S2), por ejemplo, se habilitan en la dimensión espacial 100. Para cada resolución espacial y velocidad de tramas específicas, se pueden añadir capas SNR (Q1) en la dimensión SNR 104 para mejorar la calidad de la imagen.

25 **[0075]** Una vez que el contenido de vídeo se ha codificado de dicha manera escalable, se puede usar una herramienta de extracción para adaptar el contenido real entregado de acuerdo con los requisitos de la aplicación, que pueden depender, por ejemplo, de los clientes o del canal de transmisión. En el ejemplo mostrado en la FIG. 2, cada cubo contiene imágenes con la misma velocidad de fotogramas (nivel temporal), resolución espacial y capas SNR. Se puede lograr una mejor representación añadiendo cubos (es decir, imágenes) en cualquiera de las dimensiones 100, 102 o 104. La escalabilidad combinada recibe soporte cuando hay dos, tres o incluso más escalabilidades habilitadas.

30 **[0076]** En las normas de codificación de vídeo escalables, tales como la extensión SVC a H.264 o SHVC, las imágenes con la capa espacial y SNR más baja son compatibles con el códec de vídeo de una sola capa, y las imágenes en el nivel más bajo temporal forman la capa base temporal, que se puede mejorar con imágenes en niveles temporales superiores. Además de la capa base, se pueden añadir varias capas de mejora espacial y/o de SNR para proporcionar escalabilidades espacial y/o de calidad. Cada capa de mejora espacial o de SNR puede ser temporalmente escalable, con la misma estructura de escalabilidad temporal que la capa base. Para una capa de mejora espacial o de SNR, la capa inferior de la que depende también puede denominarse capa base de esa capa de mejora espacial o de SNR específica.

40 **[0077]** La FIG. 3 es una ilustración conceptual que muestra una estructura 110 de ejemplo de un flujo de bits de codificación de vídeo escalable. La estructura de flujo de bits 110 incluye una capa 0 112 que incluye imágenes o fragmentos 10, P4 y P8, y una capa 1 114 que incluye imágenes o fragmentos B2, B6 y B10. Además, la estructura de flujo de bits 110 incluye una capa 2 116 y una capa 3 117 que incluyen cada una imágenes 0, 2, 4, 6, 8 y 10, y una capa 4 118 que incluye imágenes 0 a 11.

45 **[0078]** Una capa base tiene la capa espacial y la calidad más baja (es decir, imágenes en la capa 0 112 y de la capa 1 114 con resolución QCIF). Entre ellas, esas imágenes del nivel temporal más bajo forman la capa base temporal, como se muestra en la capa 0 112 de la FIG. 3. La capa base temporal (capa 0) 112 se puede mejorar con imágenes de un nivel temporal superior, por ejemplo, la capa 1 114 con una velocidad de tramas de 15 Hz o la capa 4 118 con una velocidad de tramas de 30 Hz.

50 **[0079]** Además de la capa base 112, 114, se pueden agregar varias capas de mejora espacial y/o de SNR para proporcionar escalabilidades espacial y/o de calidad. Por ejemplo, la capa 2 116 con resolución CIF puede ser una capa de mejora espacial a la capa base 112, 114. En otro ejemplo, la capa 3 117 puede ser una capa de mejora SNR a la capa base 112, 114 y a la capa 2 116. Como se muestra en la FIG. 3, cada capa de mejora espacial o de SNR puede ser temporalmente escalable, con la misma estructura de escalabilidad temporal que la capa base 112, 114. Además, una capa de mejora puede mejorar tanto la resolución espacial como la frecuencia de tramas. Por ejemplo, la capa 4 118 proporciona una capa de mejora de resolución 4CIF, que aumenta aún más la frecuencia de tramas de 15 Hz a 30 Hz.

60 **[0080]** La FIG. 4 es una ilustración conceptual que muestra un ejemplo de unidades de acceso de codificación de vídeo escalables 120A-120E ("unidades de acceso 120") en orden de flujo bits. Como se muestra en la FIG. 4, las imágenes o fragmentos codificados en la misma instancia de tiempo son sucesivos en el orden de flujo de bits y forman una unidad de acceso en el contexto de una norma de codificación de vídeo escalable, tal como la extensión SVC a H.264 o SHVC. Las unidades de acceso 120 siguen entonces el orden de decodificación, que podría ser diferente del orden de visualización y determinada, por ejemplo, por la relación de predicción temporal entre las unidades de acceso 120.

[0081] Por ejemplo, la unidad de acceso 120A incluye la imagen 10 de la capa 0 112, la imagen 0 de la capa 2 116, la imagen 0 de la capa 3 117 y la imagen 0 de la capa 4 118. La unidad de acceso 120B incluye la imagen P4 de la capa 0 112, la imagen 4 de la capa 2 116, la imagen 4 de la capa 3 117 y la imagen 4 de la capa 4 118. La unidad de acceso 120C incluye la imagen B2 de la capa 1 114, la imagen 2 de la capa 2 116, la imagen 2 de la capa 3 117 y la imagen 2 de la capa 4 118. La unidad de acceso 120D incluye la imagen 1 de la capa 4 118, y la unidad de acceso 120E incluye la imagen 3 de la capa 4 118.

[0082] La FIG. 5 es un diagrama de bloques que ilustra un codificador 122 de SHVC de 3 capas de ejemplo. Como se ilustra en la FIG. 5, el codificador de SHVC 122 incluye un codificador de capa base 124, un primer codificador de capa de mejora 125 y un segundo codificador de capa de mejora 126. En la sintaxis de alto nivel de solo SHVC, no hay nuevas herramientas de codificación a nivel de bloque en comparación con la codificación de capa única de HEVC. En SHVC, solo se permiten cambios de sintaxis del nivel de fragmento y por encima, tal como el filtrado de imágenes o el muestreo ascendente.

[0083] Para reducir la redundancia entre las capas, muestreada por imágenes de capa de referencia colocadas para una capa inferior/base puede generarse y almacenarse en una memoria intermedia de referencia para una capa superior/de mejora de modo que la predicción intercapas se puede conseguir de la misma manera que la predicción intertramas dentro de una única capa. Como se ilustra en la FIG. 5, se genera una imagen de referencia intercapas (ILR) 128 remuestreada a partir de una imagen de referencia en el codificador 124 de la capa base y se almacena en el primer codificador de capa de mejora 125. De forma similar, una imagen ILR 129 remuestreada se genera a partir de una imagen de referencia en el primer codificador de capa de mejora 125 y se almacena en el segundo codificador de capa de mejora 126. En SHVC WD3, la imagen ILR se marca como una imagen de referencia a largo plazo para la capa de mejora. La diferencia de vector de movimiento asociada con una imagen de referencia intercapas se limita a cero.

[0084] El próximo despliegue de los dispositivos de televisión de ultra alta definición (UHDTV) y del contenido usará una gama de colores diferente que los dispositivos heredados. Específicamente, HD usa la recomendación BT.709, la Recomendación ITU-R BT.709 "Parameter values for the HDTV standards for production and international programme exchange [Valores de parámetros para las normas de HDTV para producción e intercambio internacional de programas]" Diciembre de 2010, mientras que UHDTV usará la recomendación BT.2020, la Recomendación ITU-R BT.2020 "Parameter values for UHDTV systems for production and international programme exchange [Valores de parámetros para sistemas UHDTV para producción e intercambio de programas internacionales]", abril de 2012. Una gama de colores comprende una gama completa de colores que se pueden reproducir para una imagen, por ejemplo, en una imagen, fragmento, bloque o capa de datos de vídeo. Una diferencia clave entre estos sistemas es que la gama de colores de UHDTV es significativamente mayor que la HD. Se afirma que UHDTV proporcionará una experiencia de visualización más realista, que sea consistente con otras características de UHDTV, como la alta resolución.

[0085] La FIG. 6 es un gráfico que ilustra una gama de colores de ejemplo de una secuencia de vídeo de muestra 130. Como se ilustra en la FIG. 6, la secuencia de vídeo de muestra SWG1 130 se indica como un grupo de puntos dentro de un contorno de línea de la gama de colores UHD BT.2020 132. Para los propósitos de comparación, una visión general de la gama de colores HD BT.709 134 y un esquema de la Comisión Internacional de Iluminación (CIE) -XYZ espacio de color lineal 136 superponen la secuencia de vídeo de muestra SWG1 130. Se observa fácilmente desde la FIG. 6 que la gama de colores UHD BT.2020 132 es mucho más grande que la gama de colores HD BT.709 134. Tenga en cuenta el número de píxeles en la secuencia de vídeo de muestra SWG1 130 que se encuentran fuera de la gama de colores 134 de BT.709.

[0086] La FIG. 7 es un diagrama de bloques que ilustra la conversión de la gama de colores HD BT.709 134 a la gama de colores UHD BT.2020 132. Tanto la gama de colores HD BT.709 134 como la gama de colores UHD BT.2020 132 definen representaciones de píxeles de color en componentes de luma y de croma (por ejemplo, YCbCr o YUV). Cada gama de colores define la conversión hacia y desde el espacio de color lineal CIE-XYZ 136. Este espacio de color intermedio común se puede usar para definir la conversión de los valores de luma y de croma en la gama de colores HD BT.709 134 a los valores de luma y de croma correspondientes en la gama de colores UHD BT.2020 132.

[0087] Más detalles con respecto a la gama de colores de la secuencia de muestra ilustrada en la FIG. 6 y a la conversión de la gama de colores ilustrada en la FIG. 7 se pueden encontrar en L. Kerofsky, A. Segall, S.-H. Kim, K. Misra, "Color Gamut Scalable Video Coding: New Results [Codificación de Vídeo Escalable de Gama de Colores: Nuevos resultados]", JCTVC-L0334, Ginebra, CH, 14-23 de enero de 2013 (en lo sucesivo denominado "JCTVC-L0334").

[0088] La FIG. 8 es un diagrama de bloques que ilustra un codificador escalable de gama de colores 140 que incluye una unidad de procesamiento de predicción de colores 144 que puede generar una imagen de referencia intercapas cuando una gama de colores de capa base y una gama de colores de capas de mejora son diferentes. La unidad de procesamiento de predicción de color 144 puede usarse por un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30 de la FIG. 1, para realizar la codificación de vídeo escalable en la gama de colores, en la que la gama de colores de la base y de la capa de mejora sea diferente

[0089] En el ejemplo ilustrado en la figura. 8, un bucle de codificación de capa base 142 realiza la codificación de vídeo de imágenes que incluyen datos de color en una primera gama de colores, por ejemplo, BT.709, y un bucle de codificación de capa de mejora 146 realiza la codificación de vídeo de imágenes que incluyen datos de color en una segunda gama de colores, por ejemplo, BT.2020. La unidad de procesamiento de predicción de color 144 realiza la predicción de color para asignar o convertir datos de color de una imagen de referencia de capa base en la primera gama de colores a la segunda gama de colores, y genera una imagen de referencia intercapas para la capa de mejora basándose en los datos de color asignados a la imagen de referencia de la capa base.

[0090] Para lograr una alta eficiencia de la codificación, la unidad de proceso de predicción de color 144 está configurada para realizar la predicción de color específico en la generación de intercapas de imágenes de referencia. Como se describe con más detalle a continuación, la unidad de procesamiento de predicción de color 144 puede configurarse para realizar la predicción de color de acuerdo con cualquiera de los modelos de predicción lineal, del modelo de predicción lineal por tramos o del modelo de predicción de color basado en tabla de búsqueda 3D.

[0091] Un modelo de predicción lineal se propone en JCTVC-L0334, referenciado anteriormente. En general, el proceso de predicción de color del modelo de predicción lineal puede describirse como un modelo de ganancia y desplazamiento. El modelo de predicción lineal funciona en planos de color individuales. Para facilitar el cálculo de enteros, un parámetro describe el número de bits fraccionarios usados en el cálculo usando el parámetro numFractionBits. Para cada canal, se especifican una ganancia [c] y un desplazamiento [c]. El modelo de predicción lineal se define de la siguiente manera:

$$\text{Pred}[c][x][y] = (\text{ganancia}[c] * \text{In}[x][y] + (1 \ll (\text{numFractionBits} - 1))) \gg \text{numFractionBits} + \text{desplazamiento}[c]$$

[0092] Un modelo de predicción lineal por tramos se propone en C. Auyeung, K. Sato, "AHG14: Color gamut scalable video coding with piecewise linear predictions and shift-offset model [AHG14: Codificación de vídeo escalable de la gama de colores con predicciones lineales por tramos y modelo de desplazamiento de desplazamiento]", JCTVC-N0271, Viena, Austria, julio de 2013, basado en JCTVC-L0334, mencionado anteriormente. El proceso de predicción de color del modelo de predicción lineal por tramos también se puede describir como un modelo de ganancia y de desplazamiento. El modelo de predicción lineal por tramos se define de la siguiente manera:

$$\text{Sea } d[c][x][y] = \text{En } [c][x][y] - \text{nudo}[c].$$

Si $d[c][x][y] \leq 0$

$$\text{Pred}[c][x][y] = (\text{ganancia1}[c] * d[c][x][y] + \text{desplazamiento}[c] + (1 \ll (\text{numFractionBits} - 1))) \gg$$

numFractionBits

en caso contrario

$$\text{Pred } [c] [x] [y] = (\text{ganancia2}[c] * d[c][x][y] + \text{desplazamiento}[c] + (1 \ll (\text{numFractionBits} - 1))) \gg \text{numFractionBits}.$$

Los parámetros de predicción nudo[c], desplazamiento[c], ganancia1[c] y ganancia2[c] pueden codificarse en el flujo de bits.

[0093] Las FIGS. 9(a) y 9(b) son ilustraciones conceptuales que muestran un ejemplo de tabla de búsqueda 3D para la escalabilidad de gama de colores. Un modelo de predicción de color basado en la tabla de búsqueda 3D se propone en P. Bordes, P. Andrivon, F. Hiron, "AHG14: Color Gamut Scalable Video Coding using 3D LUT: New Results [AHG14: Codificación de vídeo escalable de gama de colores que usa 3D LUT: Nuevos resultados]", JCTVC-N0168, Viena, Austria, julio de 2013 (en lo sucesivo denominado "JCTVC-N0168"). El principio de la tabla de búsqueda 3D para la escalabilidad de gama de colores se muestra en las FIGS. 9(a) y 9(b). La tabla de búsqueda 3D 150 puede considerarse como una submuestra de un primer espacio de color 3D, por ejemplo, la gama de colores HD BT.709, donde cada vértice está asociado con un triplete de color (y, u, v) correspondiente a valores de un segundo espacio de color 3D (es decir, predichos), por ejemplo, gama de colores UHD BT.2020).

[0094] En general, la primera gama de colores puede particionarse en octantes o cuboides en cada dimensión de color (es decir, Y, U, y V), y los vértices de los octantes están asociados con el triplete de color correspondiente a la segunda gama de colores y usado para poblar la tabla de búsqueda 3D 150. El número de vértices o segmentos en cada dimensión de color indica el tamaño de la tabla de búsqueda 3D. La FIG. 9(a) ilustra los vértices o los puntos de red de intersección de los octantes en cada dimensión de color. La FIG. 9(b) ilustra los diferentes valores de color

asociados con cada uno de los vértices. Como se ilustra, en la FIG. 9(a) cada dimensión de color tiene cuatro vértices y en la FIG. 9(b) cada dimensión de color incluye cuatro valores de color.

5 **[0095]** La FIG. 10 es una ilustración conceptual que muestra la interpolación trilineal con una tabla de búsqueda 3D 152 para la escalabilidad de gama de colores. Para una muestra de color de la capa base en la primera gama de colores, el cálculo de su predicción en la segunda gama de colores para una capa de mejora se realiza mediante la interpolación trilineal de acuerdo con la siguiente ecuación:

$$\overline{\text{valor}_y} = K \times \sum_{i=0,1} \sum_{j=0,1} \sum_{k=0,1} s_i(y) \times s_j(u) \times s_k(v) \times \text{LUT}[y_i][u_j][v_k]. y$$

10 Donde:

$$K = \frac{1}{(y_1 - y_0) \times (u_1 - u_0) \times (v_1 - v_0)}$$

15 $S_0(y) = y_1 - y$ y $S_1(y) = y - y_0$

y_0 es el índice del vértice submuestreado más cercano inferior a y ,

y_1 es el índice del vértice submuestreado más cercano superior a y .

20 Más detalles de la tabla de búsqueda 3D ilustrada en la FIG. 9 y la interpolación trilineal con la tabla de búsqueda 3D ilustrada en la FIG. 10 se puede encontrar en JCTVC-N0168, mencionado anteriormente.

25 **[0096]** La FIG. 11 es una ilustración conceptual que muestra la interpolación tetraédrica con una tabla de búsqueda 3D 154 para la escalabilidad de gama de colores. La interpolación tetraédrica se puede usar en lugar de la interpolación trilineal descrita anteriormente para reducir la complejidad computacional de la tabla de búsqueda 3D.

30 **[0097]** La FIG. 12 es una ilustración conceptual que muestra seis ejemplos de tetraedros 156A-156F (colectivamente "tetraedros 156") usados para abarcar un punto P de una tabla de búsqueda 3D que se vaya a interpolar usando la interpolación tetraédrica. En el ejemplo de la FIG. 12, hay seis opciones para determinar el tetraedro que contiene el punto P que se va a interpolar en un octante de la tabla de consulta 3D dado que los vértices P₀ y P₇ deben incluirse en el tetraedro. Usando la interpolación tetraédrica, la tabla de búsqueda 3D puede diseñarse para una decisión rápida en lugar de verificar la relación de cada uno de los dos componentes: y u, y v, u y v .

35 **[0098]** En algunos ejemplos, una tabla de búsqueda 3D independiente se puede generar para cada uno de los componentes de color, es decir, el componente a luma (Y), un primer componente de croma (U) y un segundo componente de croma (V). Cada una de las tablas de búsqueda 3D incluye una dimensión de luma (Y), una primera dimensión de croma (U) y una segunda dimensión de croma (V), y se indexa usando los tres componentes de color independientes (Y, U, V).

40 **[0099]** En un ejemplo, una función de asignación se puede definir para cada componente de color basado en la tabla de búsqueda 3D. Una función de asignación de ejemplo para un valor de píxel de luma (Y) se presenta en la siguiente ecuación:

$$Y_E = \text{LUT}_Y(Y_B, U_B, V_B) * Y_B + \text{LUT}_U(Y_B, U_B, V_B) * U_B + \text{LUT}_V(Y_B, U_B, V_B) * V_B + \text{LUT}_C(Y_B, U_B, V_B)$$

45 En la ecuación anterior, Y_E representa el valor de píxel de luma en la capa de mejora, (Y_B, U_B, V_B) representa un valor de píxel de capa base, y LUT_Y , LUT_U , LUT_V y LUT_C representan la tabla de búsqueda 3D para cada componente de color Y, U, V y una constante, respectivamente. Se pueden definir funciones de asignación similares para un primer valor de píxel de croma (U) y un segundo valor de píxel de croma (V) en la capa de mejora.

50 **[0100]** En general, la tabla de búsqueda 3D de color basada en la gama de escalabilidad da como resultado un buen rendimiento de codificación. Sin embargo, el tamaño de la tabla de búsqueda 3D puede ser preocupante, ya que la tabla de búsqueda 3D se almacena en general en la memoria caché en una implementación de hardware. Convencionalmente, las tablas de búsqueda 3D siempre son simétricas, de manera que las tablas de búsqueda 3D tienen el mismo tamaño para el componente de luma, el primer componente de croma y el segundo componente de croma. Además, de manera convencional, las tablas de búsqueda 3D siempre están equilibradas de manera que un tamaño de cada dimensión de las tablas de búsqueda 3D es siempre el mismo. Esto da como resultado tamaños grandes de tablas con alta complejidad computacional y altos costes de señalización. Por ejemplo, los tamaños de tablas pueden ser de hasta 9x9x9 o 17x17x17.

60 **[0101]** En algunos casos, el tamaño de la tabla de búsqueda 3D usada para la escalabilidad de gama de colores es demasiado grande, lo que puede ocasionar dificultades en implementaciones prácticas. Además, el gran tamaño de

la tabla y el uso de la interpolación trilineal para la tabla de búsqueda 3D dan como resultado una alta complejidad computacional.

5 **[0102]** En la Solicitud de Patente de EE. UU 14/512,177 (Despacho de abogados No. 1212-712US01/140193), presentada el 10 de octubre de 2014, se proponen los siguientes procedimientos para reducir el coste de señalización y la complejidad computacional de la escala de colores basada en la tabla de búsqueda 3D.

10 **[0103]** El primer procedimiento incluye la generación de una tabla de búsqueda 3D asimétrica de manera que los componentes de luma (Y) y de croma (U y V) tienen diferentes tamaños. En algunos casos, la tabla de búsqueda 3D puede tener un tamaño más grande, es decir, más segmentos u octantes, para el componente de luma que para cada uno de los primer y segundo componentes de croma. En este caso, los componentes de croma pueden usar una tabla de búsqueda más gruesa y el componente de luma puede usar una tabla de búsqueda más refinada. Por ejemplo, los tamaños de tablas pueden ser de hasta 8x2x2. En otros casos, la tabla de búsqueda 3D puede tener un tamaño mayor para uno o ambos componentes de croma que para el componente de luma

15 **[0104]** El segundo procedimiento incluye la generación de una tabla de búsqueda 3D desequilibrada, es decir, la tabla[M][N][K], de manera que el tamaño de cada dimensión es diferente dependiendo de qué componente de color se está usando como un índice de la tabla para la tabla de búsqueda 3D. La tabla de búsqueda 3D puede tener un tamaño mayor para la dimensión asociada con el componente de color usado como índice de la tabla. En este caso, la asignación de color puede ser más precisa para el componente de color usado como índice de la tabla, mientras que es menos precisa para los otros componentes de color.

20 **[0105]** El tercer procedimiento incluye generar sólo una tabla de búsqueda 3D de componente de luma, y sólo el uso de la tabla de búsqueda 3D para realizar la predicción de componente de luma. Se pueden usar las técnicas de asignación lineal unidimensional (1D) o de asignación lineal por tramos para los componentes de croma.

25 **[0106]** Las técnicas de esta divulgación están dirigidas a la señalización de la información usada para generar tablas de búsqueda 3D para la escalabilidad de gama de colores. De acuerdo con las técnicas, el codificador de vídeo 20 puede codificar información de partición y/o valores de color de una tabla de búsqueda 3D generada para la escalabilidad de gama de colores. El decodificador de vídeo 30 puede decodificar la información de partición y/o los valores de color para generar la tabla de búsqueda 3D con el fin de realizar la escalabilidad de gama de colores. Las técnicas divulgadas proporcionan una partición eficiente de los componentes de color de la tabla de búsqueda 3D y una señalización eficiente de la información de partición y/o los valores de color para la tabla de búsqueda 3D. De esta forma, las técnicas divulgadas pueden reducir tanto el coste de señalización como la complejidad computacional para generar la tabla de búsqueda 3D. Las técnicas descritas en esta divulgación pueden ser particularmente útiles para señalar la información usada para generar tablas de búsqueda 3D asimétricas y/o desequilibradas.

30 **[0107]** En un ejemplo, las técnicas descritas en esta divulgación pueden proporcionar una partición más eficiente de los componentes de color de la tabla de búsqueda 3D al permitir particiones asimétricas de manera que la tabla de búsqueda 3D tenga una partición más gruesa para los primer y segundo componentes de croma (por ejemplo, Cb y Cr o U y V) y una partición más fina para un componente de luma (por ejemplo, Y). Las técnicas también pueden proporcionar una señalización más eficiente de la información de partición para la tabla de búsqueda 3D al señalar una cantidad de particiones adicionales para el componente de luma encima de un número base de particiones para la tabla de búsqueda 3D. En otro ejemplo, las técnicas pueden proporcionar una partición más eficiente de los componentes de color de la tabla de búsqueda 3D al permitir la partición conjunta de los primer y segundo componentes de croma (por ejemplo, Cb y Cr o U y V).

35 **[0108]** Las técnicas también pueden proporcionar una señalización más eficiente de la información usada para generar la tabla de búsqueda 3D para la escalabilidad de gama de colores (CGS) al habilitar uno o más de los siguientes. En un primer ejemplo, se puede señalar un indicador o un índice para indicar dónde se señala la información de predicción de color CGS, tal como en un conjunto de parámetros de vídeo (VPS), un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS), un encabezado de sector o cualquier otro encabezado de sintaxis de alto nivel. En un segundo ejemplo, se pueden señalar varias particiones para indicar un tamaño, es decir, un número de segmentos u octantes, de una tabla de búsqueda 3D asimétrica y/o desequilibrada. En un tercer ejemplo, cuando los componentes de croma se particionan conjuntamente, se puede señalar un rango de una partición de centro de croma.

40 **[0109]** En un cuarto ejemplo, los parámetros de nivel inferior (por ejemplo, nivel de fragmento) de la información de predicción de color CGS puede codificarse de manera predictiva a partir de los parámetros de nivel superior (por ejemplo, nivel de PPS) de color CGS. En un quinto ejemplo, una tabla de sintaxis de la información de predicción de color CGS puede señalizarse en el flujo de bits, tal como en el VPS, en el SPS, en el PPS o en el encabezado de fragmento. Cuando se señalizan varias tablas de sintaxis de predicción de color CGS en diferentes ubicaciones en el flujo de bits, la tabla de sintaxis en el nivel más bajo que cubre la imagen que se vaya a codificar puede usarse para la imagen. En un sexto ejemplo, la tabla de sintaxis de la información de predicción de color CGS puede señalizarse condicionalmente según si la predicción de textura está habilitada para la codificación de la imagen. Cuando una capa superior de datos de vídeo, es decir, una capa de mejora, tiene múltiples capas de referencia de textura, las tablas de

sintaxis de predicción de color CGS pueden señalizarse para todas o algunas de las capas de referencia cuya gama de colores sea diferente de la capa de mejora. En un séptimo ejemplo, para mantener la baja complejidad, la tabla de sintaxis de predicción de color CGS se puede restringir aún más para que se señalice, como máximo, para solo una capa de referencia por imagen.

5 [0110] En un octavo ejemplo, un límite de partición puede señalizarse por al menos uno de los primer y segundo componentes de croma con el fin de obtener la partición desigual a lo largo de la una de las direcciones de croma en la tabla de búsqueda 3D. La información de los límites de partición puede señalizarse condicionalmente cuando el al menos uno de los componentes de croma se particione en dos o más segmentos u octantes a lo largo de la dirección de croma.

10 [0111] Una vez que cada uno de los componentes de color de la tabla de búsqueda 3D se particiona en uno o más octantes, las técnicas descritas en la presente divulgación pueden proporcionar señalización más eficiente de los valores de color de la tabla de búsqueda 3D. Las técnicas incluyen señalar, para cada octante para cada uno de los componentes de color de la tabla de búsqueda 3D, ya sean valores de vértices de cada uno de los octantes o coeficientes de asignación de color de una función de asignación de color lineal para cada uno de los octantes. En esta divulgación, los términos "partición", "octante", "segmento" y "cuboide" se pueden usar indistintamente para describir las regiones particionadas de los componentes de color de la tabla de búsqueda 3D.

15 [0112] En un primer ejemplo, para cada octante para cada uno de los componentes de color, pueden señalizarse vértices del octante. En este ejemplo, se puede señalar un valor residual entre un valor predicho de un vértice dado y un valor real del vértice dado. En cualquier caso, el valor residual de predicción puede cuantizarse. La información de etapa de cuantización, por ejemplo, un valor de cuantización, puede señalizarse o puede ser un valor predefinido. El valor residual puede codificarse con la codificación exp-golomb de orden k. El orden k puede señalizarse en el flujo de bits o derivarse adaptativamente en función de otra información, tal como la magnitud de los valores residuales, en el flujo de bits. Para cada octante o partición, no todos los vértices necesitan señalizarse. Por ejemplo, al menos cuatro vértices pueden señalizarse si los octantes o cuboides adyacentes no comparten valores de vértice. Los al menos cuatro vértices se pueden usar para interpolar todos los valores en el octante o cuboide.

20 [0113] En un segundo ejemplo, para cada octante para cada uno de los componentes de color, pueden señalizarse coeficientes de asignación de color (es decir, a, b, c y d) para una función de asignación de color lineal de valores de color en la tabla de búsqueda 3D en vez de los vértices del octante. La función de asignación de color lineal con parámetros de asignación de color se puede usar directamente para realizar la predicción de la gama de colores. La función de asignación de color lineal se usa para convertir datos de color en una primera gama de colores para una capa inferior de datos de vídeo en una segunda gama de colores para una capa superior de datos de vídeo, y los coeficientes de asignación de color son factores de ponderación entre los componentes de color de la parte inferior y capas superiores de los datos de vídeo. En esta divulgación, los términos "coeficientes de asignación de color" y "coeficientes de predicción de color lineal" se pueden usar indistintamente. Además, los términos "función de asignación de color lineal", "función de predicción de color lineal" y "ecuación lineal 3D" también se pueden usar de manera intercambiable.

25 [0114] En este ejemplo, los coeficientes de asignación de color (es decir, a, b, c y d) se pueden convertir o cuantizarse desde valores de punto flotante en un valor entero usando un número predefinido de bits. En algunos casos, la información de conversión o cuantización puede señalizarse en el flujo de bits. En otros casos, la información de conversión o cuantización (es decir, el número de bits usados para representar el valor de 1) puede depender de al menos una de la profundidad de bits de entrada o de la profundidad de bits de salida de la tabla de búsqueda 3D.

30 [0115] Para cada uno de los componentes de color, uno de los coeficientes de asignación de color de la función de asignación de color lineal puede ser un coeficiente clave que defina un factor de ponderación del mismo componente de color que se esté prediciendo. Por ejemplo, al predecir el primer componente de croma de la capa superior (es decir, U_e) usando la función de asignación de color lineal $U_e = a \cdot Y_b + b \cdot U_b + c \cdot V_b + d$, b es el coeficiente clave porque es el factor de ponderación entre el primer componente de croma de la capa inferior (es decir, U_b) y el primer componente de croma de la capa superior (es decir, U_e) que se predice. La señalización del coeficiente clave puede ser diferente de los otros coeficientes. En algunos ejemplos, la predicción de los coeficientes clave puede depender de un valor predefinido distinto de cero, mientras que la predicción de los otros coeficientes puede depender de un valor predicho igual a cero. En otros ejemplos, la predicción de los coeficientes clave puede depender de al menos uno de la profundidad de bits de entrada o de la profundidad de bits de salida de la tabla de búsqueda 3D.

35 [0116] Los numerosos ejemplos descritos anteriormente de técnicas para la partición eficiente y la señalización de una tabla de búsqueda 3D para la escalabilidad de gama de colores pueden usarse solo o en cualquier combinación, y no debe limitarse a las combinaciones de ejemplo descritas en esta divulgación. A continuación se proporcionan detalles adicionales de las técnicas descritas para la partición eficiente de los componentes de color de la tabla de búsqueda 3D y la señalización eficiente de la información de la partición y/o los valores de color para la tabla de búsqueda 3D.

65

[0117] Como se describió anteriormente, en un ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede generar una tabla de búsqueda 3D para la escalabilidad de gama de colores mediante la realización de la partición conjunta de los primer y segundo componentes de croma. En una tabla de búsqueda 3D convencional, los componentes de luma, de primer croma y de segundo croma (es decir, Y, U y V) se particionan de forma independiente. Cuando cada componente se divide en N segmentos u octantes, el número total de octantes puede ser NxNxN, lo que da como resultado una gran tabla de búsqueda 3D. Por ejemplo, los tamaños de tablas pueden ser de hasta 9x9x9 o 17x17x17. Para reducir el número de octantes, las técnicas de esta divulgación pueden proporcionar una partición independiente del componente de luma (es decir, Y) mientras se particionan conjuntamente los primer y segundo componentes de croma (es decir, U y V).

[0118] Por ejemplo, el componente de luma puede dividirse uniformemente en M particiones u octantes. El espacio 2D UxV de los primer y segundo componentes de croma se puede dividir en dos particiones de la siguiente manera:

$$\text{si } (2^{CBit-1} - R < u < 2^{CBit-1} + R \text{ y } 2^{CBit-1} - R < v < 2^{CBit-1} + R)$$

$$(u, v) \rightarrow \text{Partición 0}$$

en caso contrario

$$(u, v) \rightarrow \text{Partición 1}$$

donde (u , v) indica los valores de píxel de las componentes U y V, CBit representa la profundidad de bits de las componentes de croma, 2^{CBit-1} corresponde a un valor central de los píxeles de croma y R indica la distancia al valor central 2^{CBit-1} . En algunos casos, R puede ser un valor fijo predefinido; de lo contrario, R puede ser un valor señalado en el flujo de bits, tal como en el VPS, en el SPS, en el PPS o en el encabezado de fragmento.

[0119] La FIG. 13 es una ilustración conceptual que muestra un ejemplo de tabla de búsqueda 3D 158 con un componente de luma particionado independientemente y componentes de croma particionados conjuntamente. En el ejemplo ilustrado de la FIG. 13, el componente de luma (es decir, Y) se particiona en cuatro partes de acuerdo con las líneas de partición 160A, 160B y 160C. Los componentes de croma (es decir, UV) se particionan en dos regiones de acuerdo con una partición cuboide 162. En este caso, para un valor de píxel de par de croma (u , v), está dentro de la partición cuboide 162 o fuera de la partición cuboide 162. En el ejemplo de la FIG. 13, la tabla de búsqueda 3D 158 se particiona en particiones $4 \times 2 = 8$.

[0120] En otro ejemplo, las componentes de croma (es decir, UV) se particionan conjuntamente mientras que el componente de luma (es decir, Y) se divide en M particiones, pero las particiones M pueden no ser necesariamente del mismo tamaño. En otras palabras, el componente de luma puede particionarse de manera desigual de manera que al menos una de las particiones tenga un tamaño diferente al de las otras particiones. Por ejemplo, una partición ubicada cerca de un valor central del componente de luma puede ser más refinada, es decir, más pequeña, que las particiones ubicadas más alejadas del valor central.

[0121] En el ejemplo de partición del componente de croma conjunto (es decir, UV), las tablas de sintaxis y la semántica relacionada para señalar los coeficientes de asignación de color de una función de asignación de color lineal para la escalabilidad de gama de colores (CGS) pueden ser las siguientes en las Tablas 1-3 siguientes. Cualquier edición, adición o actualización del SHVC WD3, mencionada anteriormente, se indica con texto en *cursiva*.

TABLA 1: Sintaxis de la RBSP del conjunto de parámetros de imagen (PPS)

pic_parameter_set_rbsp() {	Descriptor
...	
pps_extension_flag	u(1)
<i>si(nuh_layer_id > 0)</i>	
cgs_enable_flag	u(1)
<i>si(nuh_layer_id > 0 && cgs_enable_flag) {</i>	
cgs_info_in_pps_flag	u(1)
<i>si (cgs_info_in_pps_flag)</i>	
<i>cgs_info_table()</i>	
}	
...	

pic_parameter_set_rbsp() {	Descriptor
}	

El **cgs_enable_flag** igual a 1 indica que la escalabilidad de gama de colores está habilitada. El **Cgs_enable_flag** igual a 0 especifica que la escalabilidad de gama de colores está deshabilitada. Cuando no está presente, se deduce que **cgs_enable_flag** es 0.

5 El **cgs_info_in_pps_flag** igual a 1 especifica que **cgs_info_table** está presente en el PPS. **cgs_info_in_pps_flag** igual a 0 especifica que **cgs_info_table** no está presente en el PPS pero está presente en el encabezado de fragmento. Cuando no está presente, se deduce que **cgs_info_in_pps_flag** es 0.

10 **TABLA 2: Sintaxis de información de predicción de color de la escalabilidad de gama de colores (CGS)**

<i>cgs_info_table()</i> {	Descriptor
<i>cgs_uv_part_range_from_center</i>	<i>ue(v)</i>
<i>cgs_y_part_num_log2</i>	<i>ue(v)</i>
<i>para(i = 0; i < CGS_PART_NUM; i++) {</i>	
<i> para(j = 0; j < 3; j++) {</i>	
<i> para(l = 0; l < 4; l++) {</i>	
<i> si(j==l)</i>	
<i>cgs_color_pred_coeff_minus128[i][j][l]</i>	<i>se(v)</i>
<i> en caso contrario</i>	
<i>cgs_color_pred_coeff[i][j][l]</i>	<i>se(v)</i>
<i> }</i>	
<i> }</i>	
<i>}</i>	

15 El elemento sintáctico **cgs_uv_part_range_from_center** especifica el rango de partición de croma desde el valor central del componente de croma. Cuando no está presente, se deduce que **cgs_uv_part_range_from_center** es 0.

El elemento sintáctico **cgs_y_part_num_log2** especifica el número de particiones de luma en la predicción de color CGS. Cuando no está presente, se deduce que **cgs_y_part_num_log2** es 0. El parámetro **CGS_PART_NUM** se deriva de la manera siguiente: **CGS_PART_NUM = 1 << (cgs_y_part_num_log2+1)**.

20 El elemento sintáctico **cgs_color_pred_coeff_minus128** y el elemento sintáctico **cgs_color_pred_coeff** especifican cada uno los coeficientes de asignación de color de una función de asignación de color lineal para la CGS. Cuando no está presente, se deduce que son 0. Cabe señalar que, en algunos ejemplos, los elementos sintácticos **cgs_color_pred_coeff_minus128** y **cgs_color_pred_coeff** pueden señalizarse usando diferentes procedimientos de codificación por entropía. En el ejemplo de la Tabla 2 anterior, se usa el procedimiento de codificación por entropía de **se(v)**. De forma alternativa, se puede usar la codificación exp-golomb de orden k o la codificación de longitud fija. También se debe tener en cuenta que el elemento sintáctico **cgs_color_pred_coeff_minus128** puede indicar el valor predicho para un coeficiente clave como un número fijo predefinido igual a 128, que es el valor entero usado para representar un valor de punto flotante de 1,0 en este ejemplo.

30 **TABLA 3: Sintaxis de encabezado de fragmento**

slice_segment_header() {	Descriptor
...	
<i>si(nuh_layer_id > 0 && cgs_enable_flag && !cgs_info_in_pps && NumActiveRefLayerPics > 0)</i>	
<i> cgs_info_table()</i>	
<i>if(sample_adaptive_offset_enabled_flag) {</i>	
...	

slice_segment_header() {	Descriptor
}	

Quando la escalabilidad de gama de colores está habilitada (por ejemplo, `cgs_enable_flag = 1`) y la `cgs_info_table` no está presente en el PPS (por ejemplo, `cgs_info_in_pps_flag = 0`), la `cgs_info_table` no está presente en el PPS pero está presente en el encabezado de fragmento.

5 **[0122]** Como se describió anteriormente, en otro ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden generar una tabla de búsqueda 3D para la escalabilidad de gama de colores con la partición gruesa para los primer y segundo componentes de croma (por ejemplo, Cb y Cr o U y V) y con la partición más fina para el componente de luma (por ejemplo, Y). El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden generar esta tabla de
10 búsqueda 3D al particionar cada uno de los componentes de color en un número de octantes de acuerdo con un valor de partición base, por ejemplo, una profundidad de división máxima para la tabla de búsqueda 3D, y entonces la partición adicional de cada uno de los octantes del componente de luma se basa en un valor de partición de luma. En un ejemplo, el valor de partición de luma puede señalizarse en un flujo de bits por el codificador de vídeo 20 al decodificador de vídeo 30. En algunos casos, el valor de partición base también puede señalizarse en el flujo de bits
15 por el codificador de vídeo 20 al decodificador de vídeo 30. En otros casos, el valor de partición de luma se puede derivar tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30, y/o el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo como en el decodificador de vídeo.

20 **[0123]** En un ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 primero construyen la tabla de búsqueda 3D de tal manera que cada uno de los componentes de color (es decir, el espacio YUV) se divide o particiona de manera iterativa y simétrica hasta que se alcanza una profundidad dividida predefinida o señalizada. La profundidad de división define un número máximo de veces que se pueden particionar todos los componentes de color de la tabla de búsqueda 3D. De esta manera, el valor de partición base puede ser una profundidad de división predefinida.
25 Entonces, el codificador de vídeo 20 y/o el decodificador de vídeo 30 aún más uniformemente, o no uniformemente, divide cada cubo u octante más pequeño a lo largo de la dirección de luma (es decir, Y) de modo que el componente de luma (es decir, Y) tenga una partición más fina mientras que los componentes de croma (es decir, U y V) tengan particiones más gruesas.

30 **[0124]** Por ejemplo, la tabla de búsqueda 3D propuesta con una partición más fina para el componente de luma y con una partición más gruesa para los componentes de croma puede indicarse de la manera siguiente en la Tabla 4 a continuación. Cualquier edición, adición o actualización del SHVC WD3, mencionada anteriormente, se indica con texto en *cursiva*.

TABLA 4: Sintaxis de datos de color de tabla de búsqueda 3D

3D_LUT_color_data () {	Descriptor
cur_octant_depth	u(3)
cur_y_part_num_log2	u(2)
input_bit_depth_minus8	u(4)
output_bit_depth_minus8	u(4)
res_quant_bit	u(3)
coding_octant(0, 0, 0, 0, 1 << InputBitDepth)	
}	

El elemento sintáctico **cur_octant_depth** indica la profundidad de división máxima para el espacio YUV para la tabla actual. En otras palabras, el elemento sintáctico `cur_octant_depth` indica el valor de partición base para la tabla de
40 búsqueda 3D.

El elemento sintáctico **cur_y_part_num_log2** especifica el número de particiones Y para el cubo más pequeño. De forma alternativa, el elemento sintáctico `cur_y_part_num_log2` especifica el número de particiones Y para el cubo cuyo `split_octant_flag` es igual a 0. En otras palabras, el elemento sintáctico `cur_y_part_num_log2` indica el valor de partición de luma para el componente de luma de la tabla de búsqueda 3D. En un ejemplo, el valor de partición base indicado
45 por `cur_octant_depth` es igual a 1, de manera que cada uno de los componentes de color se particiona en un único octante, y el valor de partición de luma indicado por `cur_y_part_num_log2` es igual a 4 de manera que el único octante del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 4x1x1. Como otro ejemplo, el valor de partición base indicado por `cur_octant_depth` es igual a 2, de manera que cada uno de los componentes de color se particiona en dos octantes, y el valor de partición de luma indicado por

cur_y_part_num_log2 es igual a 4, de manera que cada uno de los dos octantes del componente se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 8x2x2.

5 El elemento sintáctico **input_bit_depth_minus8** especifica la profundidad de bits de las entradas de la tabla de búsqueda 3D. El parámetro InputBitDepth se puede calcular de la siguiente manera: $\text{InputBitDepth} = 8 + \text{input_bit_depth_minus8}$.

10 El elemento sintáctico **ouput_bit_depth_minus8** especifica la profundidad de bits de la salida de la tabla de búsqueda 3D. El parámetro OutputBitDepth se puede calcular de la siguiente manera: $\text{OutputBitDepth} = 8 + \text{output_bit_depth_minus8}$.

15 El elemento sintáctico **res_quant_bit** especifica el número de bits usados para cuantizar los valores residuales del vértice o los valores residuales del coeficiente de asignación de color para cada octante para cada componente de color de la tabla de búsqueda 3D. La cuantización de los valores residuales se logra desplazando a la derecha los valores residuales del vértice o los valores residuales del coeficiente de asignación de color mediante **res_quant_bit**.

20 La tabla de sintaxis coding_octant se describe con más detalle con respecto a la Tabla 5 a continuación. En el ejemplo de la tabla de sintaxis coding_octant mostrada en la Tabla 5 a continuación, solo el octante o cuboide más pequeño se divide además a lo largo del luma (es decir, dirección Y), de manera que el componente de luma (es decir, Y) tiene una partición más fina que los componentes de croma (es decir, U y V). En algunos ejemplos, cualquier octante o cuboide puede dividirse a lo largo de la dirección de luma. En este ejemplo, si un octante se divide más a lo largo de la dirección de luma se puede señalar en el flujo de bits.

25 **[0125]** Como se ha descrito anteriormente, en un ejemplo adicional, el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede generar la tabla de consulta en 3D basándose en varios octantes para cada uno de los componentes de color y de los valores de color para cada uno de los octantes. En algunos casos, el codificador de vídeo 20 al decodificador de vídeo 30 puede señalar el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D. Para que el decodificador de vídeo 30 determine los valores de color para cada octante para cada uno de los componentes de color de la tabla de búsqueda 3D, el codificador de vídeo 20 puede indicar vértices
30 de cada uno de los octantes o coeficientes de asignación de color para una función de asignación de color lineal de valores de color para cada uno de los octantes.

35 **[0126]** En un ejemplo descrito anteriormente, para cada uno de los octantes o particiones, el codificador de vídeo 20 puede señalar los coeficientes de asignación de color de la función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D. La función de asignación de color lineal se usa para convertir datos de color en una primera gama de colores para una capa inferior de datos de vídeo en una segunda gama de colores para una capa superior de datos de vídeo, y los coeficientes de asignación de color son factores de ponderación entre los componentes de color de la parte inferior y capas superiores de los datos de vídeo. Para cada uno de los componentes de color, uno de los coeficientes de asignación de color puede ser un coeficiente clave que defina un factor de ponderación entre el mismo componente de color de las capas inferior y más alta de los datos de vídeo.
40

[0127] La función de asignación de color lineal común puede representarse de la manera siguiente.

45

$$\begin{bmatrix} Y_e \\ U_e \\ V_e \end{bmatrix} = \begin{bmatrix} a_{00} \cdot Y_b + b_{01} \cdot U_b + c_{02} \cdot V_b + d_{03} \\ a_{10} \cdot Y_b + b_{11} \cdot U_b + c_{12} \cdot V_b + d_{13} \\ a_{20} \cdot Y_b + b_{21} \cdot U_b + c_{22} \cdot V_b + d_{23} \end{bmatrix}$$

50 En esta función de ejemplo, el subíndice e y b indica la capa superior (es decir, la capa de mejora) y la capa inferior (por ejemplo, la capa base), respectivamente, para cada uno de los componentes de color de luma, de primer croma y de segundo croma (es decir, Y, U, y V). Los parámetros a, b, c y d representan los coeficientes de asignación de color. En algunos ejemplos, los coeficientes de asignación de color a_{00} , b_{11} y c_{22} representan los coeficientes clave para cada uno de los componentes de color, es decir, los factores de ponderación entre el mismo componente de color de la base y las capas de mejora. Aunque estos coeficientes se conocen como coeficientes clave en esta divulgación, este nombre no debe considerarse limitativo, ya que otros nombres pueden hacer referencia a coeficientes definidos de manera similar. En algunos ejemplos, los coeficientes de asignación de color (es decir, a, b, c y d) de la función de asignación de color lineal para un octante dado pueden convertirse primero en los vértices del octante dado, y entonces
55 los valores de los vértices pueden codificarse en el flujo de bits para representar los valores de color en la tabla de búsqueda 3D.

60 **[0128]** En algunos ejemplos, los coeficientes de asignación de color (es decir, a, b, c, y d) de la función de asignación de color lineal se derivan como valores de punto flotante. En este ejemplo, el codificador de vídeo 20 puede convertir o cuantizar los valores de punto flotante de los coeficientes de asignación de color en valores enteros, y entonces codificar los valores enteros en el flujo de bits para cada octante. Por ejemplo, los valores enteros de los coeficientes de asignación de color se pueden codificar en la cgs_info_table representada en la Tabla 2 anterior, o se pueden

codificar en la tabla coding_octant representada en la Tabla 5 a continuación. El decodificador de vídeo 30 puede realizar operaciones de enteros usando los valores enteros de los coeficientes de asignación de color.

[0129] Para representar los valores de punto flotante de los coeficientes de asignación de color con una precisión razonable, se selecciona un valor entero para representar un valor de punto flotante de 1,0, por ejemplo, usando 256 (8 bits) como valor entero para representar el valor flotante valor en puntos de 1,0. El codificador de vídeo 20 puede realizar la conversión o cuantización de acuerdo con la siguiente ecuación: $A = L \cdot a \cdot 2^N$, donde a indica el valor de punto flotante del coeficiente de asignación de color que se vaya a convertir o cuantizar, A es el valor entero convertido o cuantizado, $\lfloor x \rfloor$ indica una función de suelo que redondea un parámetro x a un valor entero máximo que es menor que x , y N indica una cantidad de bits necesarios para convertir o cuantizar el valor de punto flotante de 1,0. De esta forma, los valores enteros que representan los valores de punto flotante tienen una profundidad de bits (por ejemplo, 8 bits) basándose en el parámetro N .

[0130] En un ejemplo, la conversión o la cuantización puede basarse en el parámetro N , en el exponente de la ecuación anterior $A = L \cdot a \cdot 2^N$, establecido en un valor fijo predefinido, tal como 8 o 10. En otro ejemplo, la conversión o cuantización puede basarse en un valor de N determinado basándose al menos en una profundidad de bits de entrada (es decir, B_i) o una profundidad de bits de salida (es decir, B_o) de la tabla de búsqueda 3D. Por ejemplo, la conversión o la cuantización puede basarse en el parámetro N determinado de acuerdo con una de las siguientes ecuaciones.

$$N = B_i,$$

$$N = B_o,$$

$$N = 10 + B_i - B_o, \text{ o}$$

$$N = 8 + B_i - B_o.$$

[0131] En algunos ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede predecir los coeficientes de asignación de color, y los valores residuales de código de los coeficientes de asignación de color como la diferencia entre los valores originales de los coeficientes de asignación de color y los valores predichos de los coeficientes de asignación de color. Por ejemplo, para un octante dado, la predicción o parte de la predicción para al menos uno de los coeficientes de asignación de color, por ejemplo, uno de los coeficientes clave, puede basarse en un valor predicho igual a un valor fijo predefinido. En un ejemplo, el valor predicho se puede establecer igual a 2^N , donde N es el valor del bit de cuantización descrito anteriormente. Como otro ejemplo, para el octante dado, la predicción o parte de la predicción para al menos uno de los coeficientes de asignación de color, por ejemplo, uno de los coeficientes clave, puede depender de al menos una de la profundidad de bits de entrada (es decir, B_i) o de la profundidad de bits de salida (es decir, B_o) de la tabla de búsqueda 3D. Por ejemplo, la predicción o parte de la predicción se puede basar en un valor predicho igual a $2^{N+B_o-B_i}$.

[0132] Como ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden realizar la predicción de los coeficientes de asignación de color de la siguiente manera. Para un primer octante para cada uno de los componentes de color, los coeficientes de asignación de color de la función de asignación de color lineal pueden predecirse basándose en valores fijos predefinidos. El coeficiente clave para cada uno de los componentes de color puede predecirse de manera diferente a los otros coeficientes. Por ejemplo, un coeficiente clave puede predecirse basándose en un valor predicho igual a un valor predefinido distinto de cero, y cualquier coeficiente de asignación de color restante puede predecirse basándose en un valor predicho igual a cero. En este ejemplo, los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color se pueden predecir basándose en los valores predichos de al menos un octante anterior, tal como el primer octante.

[0133] Como otro ejemplo de la predicción de los coeficientes de asignación de color, para el primer octante para cada uno de los componentes de color, el valor de predicción para los coeficientes clave para todos los componentes de color puede fijarse igual a $2^{N+B_o-B_i}$ y los valores de predicción para los otros coeficientes se pueden establecer iguales a 0. En este ejemplo, los coeficientes de los octantes restantes para cada uno de los componentes de color se pueden predecir a partir del octante anterior. En un ejemplo adicional, la predicción de los coeficientes de asignación de color se puede realizar entre diferentes particiones u octantes para cada uno de los componentes de color. De forma alternativa, un conjunto de coeficientes de asignación de color puede señalarse como coeficientes base, tal como en el SPS o en el PPS. Entonces, las diferencias entre los valores reales del coeficiente de asignación de color y los valores de los coeficientes base pueden señalarse a nivel de imagen o fragmento.

[0134] En algunos casos, los valores residuales de los coeficientes de asignación de color pueden cuantizarse basándose en un valor de cuantización determinado. El codificador de vídeo 20 puede señalar el valor de cuantización determinado para que el decodificador de vídeo 30 realice una cuantización inversa para decodificar

adecuadamente los coeficientes de asignación de color. En un ejemplo, el valor de cuantización determinado se puede indicar mediante el elemento sintáctico `res_quant_bit` descrito con más detalle con respecto a la Tabla 4 anterior.

5 **[0135]** En este caso, para cada uno de los octantes para cada uno de los componentes de color, el codificador de vídeo 20 puede calcular los valores residuales de los coeficientes de asignación de color basándose en los valores originales de los coeficientes de asignación de color y en los valores previstos de los coeficientes de asignación de color, cuantizar los valores residuales de los coeficientes de asignación de color basándose en el valor de cuantización determinado, y entonces cuantizar los valores residuales de los coeficientes de asignación de color en el flujo de bits. El codificador de vídeo 20 también puede codificar el elemento sintáctico `res_quant_bit` para indicar el valor de
10 cuantización determinado. El decodificador de vídeo 30 decodifica entonces el elemento sintáctico `res_quant_bit` y los valores residuales de los coeficientes de asignación de color, cuantifica de manera inversa los valores residuales de los coeficientes de asignación de color basados en el valor de cuantización determinado, y reconstruye los coeficientes de asignación de color basándose en los valores residuales decodificados y en los valores predichos de los coeficientes de asignación de color.

15 **[0136]** Además, los valores de los coeficientes de asignación de color pueden restringirse para estar dentro de un rango dado basándose en un valor fijo predefinido o en un valor dependiente de al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D. El valor de los coeficientes de asignación de color (es decir, a, b, c y d) puede estar limitado a un cierto rango para reducir la complejidad computacional de generar la tabla de búsqueda 3D. Como ejemplo, se puede restringir el valor para que esté en el
20 rango de -2^M a 2^M-1 , inclusive, donde M se establece igual a un valor fijo predefinido, tal como 10 o 12. De forma alternativa, el valor de M puede depender de uno o más del valor de bit de cuantización N , de la profundidad de bit de entrada (es decir, B_i) o de la profundidad de bit de salida (es decir, B_o) de la tabla de consulta 3D.

25 **[0137]** En otro ejemplo descrito anteriormente, para cada uno de los octantes o particiones, el codificador de vídeo 20 puede señalar valores de vértices del octante para indicar los valores de color en la tabla de búsqueda 3D. La tabla de sintaxis `coding_octant`, que se puede usar para señalar los valores de color de la tabla de búsqueda 3D, se describe principalmente en esta divulgación con respecto a los valores de señalización de vértices octantes. Sin embargo, esta descripción no debe interpretarse como limitante, ya que una tabla de sintaxis `coding_octant`
30 sustancialmente similar puede usarse para señalar valores de coeficientes de asignación de color para una función de asignación de color lineal para cada octante.

35 **[0138]** La tabla de sintaxis `coding_octant`, incluida en la sintaxis de datos de color de la tabla de búsqueda 3D mostrada en la Tabla 4 anterior, se describe con respecto a la Tabla 5 a continuación. Cualquier edición, adición o actualización del SHVC WD3, mencionada anteriormente, se indica con texto en *cursiva*.

TABLA 5: Sintaxis de octante de codificación

octante de codificación (profundidad, y, u, v, longitud) {	Descriptor
<i>si (profundidad < car_octant_depth)</i>	
<i>split_octant_flag</i>	<i>u(1)</i>
<i>si(split_octant_flag) {</i>	
<i>para(l = 0; l < 2; l++)</i>	
<i>para(m = 0; m < 2; m++)</i>	
<i>para(n = 0; n < 2; n++)</i>	
<i>coding_octant(profundidad+1, y+l*longitud/2, u+m*longitud/2, v+n*longitud/2, longitud/2)</i>	
<i>}</i>	
<i>por el contrario {</i>	
<i>para(i = 0 ; i < YPartNum ; i++)</i>	
<i>para(vértice = 0; vértice < 4; vértice++) {</i>	
<i>encoded_vertex_flag</i>	<i>u(1)</i>
<i>si(encoded_vertex_flag){</i>	
<i>res_Y[yldx][uldx][vldx][vértice]</i>	<i>se(v)</i>
<i>resU[yldx][uldx][vldx][vértice]</i>	<i>se(v)</i>
<i>res_V[yldx][uldx][vldx][vértice]</i>	<i>se(v)</i>

octante de codificación (profundidad, y, u, v, longitud) {	Descriptor
}	
}	
}	
}	

El **split_octant_flag** igual a 1 especifica que un octante se divide en ocho octantes con la mitad del tamaño en todas las direcciones para el propósito de la codificación de octante residual de vértice. Cuando no está presente, se deduce que es 0.

5

La variable YPartNum se deriva como $YPartNum = 1 \ll cur_y_part_num_log2$.

El **encoded_vertex_flag** igual a 1 especifica que los residuos del vértice con índice $[yIdx+1][uIdx][vIdx]$ [vértice] están presentes. El **encoded_vertex_flag** igual a 0 especifica que los residuos para el vértice no están presentes. Cuando no está presente, se deduce que el indicador es igual a 0.

10

La variable yIdx se deriva de la manera siguiente.

$$yIdx = (y + 1 * (\text{length} \gg cur_y_part_num_log2)) \gg (\text{InputBitDepth} - cur_octant_depth - cur_y_part_num_log2)$$

15

La variable uIdx se deriva de la manera siguiente.

$$uIdx = u \gg (\text{InputBitDepth} - cur_octant_depth)$$

20

La variable vIdx se deriva de la manera siguiente.

$$vIdx = v \gg (\text{InputBitDepth} - cur_octant_depth)$$

resY $[yIdx][uIdx][vIdx]$ [vértice], **resU** $[yIdx][uIdx][vIdx]$ [vértice] y **resV** $[yIdx][uIdx][vIdx]$ [vértice] son las diferencias (es decir, valores residuales) entre los componentes Y, U y V del vértice con índice $[yIdx][uIdx][vIdx]$ [vértice] y los valores predichos de los componentes Y, U y V para este vértice. Cuando no están presentes, se deduce que estas diferencias de **resY** $[yIdx][uIdx][vIdx]$ [vértice], **resU** $[yIdx][uIdx][vIdx]$ [vértice] y **resV** $[yIdx][uIdx][vIdx]$ [vértice] son iguales a 0.

25

[0139] En la técnica de señalización de ejemplo de coeficientes de asignación de color para una función de asignación de color lineal para cada octante de la tabla de búsqueda 3D, la tabla de sintaxis coding_octant puede indicar valores residuales que sean las diferencias entre los coeficientes de asignación de color (es decir, a, b, c, y d) para la función de asignación lineal de colores del octante y los valores del coeficiente de asignación de color predichos para el octante, en lugar de señalar los valores residuales de vértice **resY** $[yIdx][uIdx][vIdx]$ [vértice], **resU** $[yIdx][uIdx][vIdx]$ [vértice] y **resV** $[yIdx][uIdx][vIdx]$ [vértice].

35

[0140] Volviendo a la técnica de ejemplo de los valores de señalización de vértices octantes, cada entrada de la tabla de búsqueda 3D se puede derivar de la siguiente manera:

$$lutX[yIdx][uIdx][vIdx][vertex] = (resX[yIdx][uIdx][vIdx][vertex] \ll res_quant_bit) + predX[yIdx][uIdx][vIdx][vertex],$$

40

donde X indica cada uno de los componentes de color Y, U y V y $predX[yIdx][uIdx][vIdx]$ [vértice] se deriva de acuerdo con la Tabla 6 a continuación.

TABLA 6: Valores predichos para vértices de octantes en la tabla de búsqueda 3D

45

$[yIdx][uIdx][vIdx]$ [vértice]	vértice = 0	vértice = 1	vértice = 2	vértice = 3
$predY[yIdx][uIdx][vIdx]$ [vértice]	$yIdx \ll yoShift$	$yIdx \ll yoShift$	$yIdx \ll yoShift$	$(yIdx + 1) \ll yoShift$
$predU[yIdx][uIdx][vIdx]$ [vértice]	$uIdx \ll uoShift$	$(uIdx + 1) \ll uoShift$	$(uIdx + 1) \ll uoShift$	$(uIdx + 1) \ll uoShift$
$predV[yIdx][uIdx][vIdx]$ [vértice]	$vIdx \ll voShift$	$vIdx \ll voShift$	$(vIdx + 1) \ll voShift$	$(vIdx + 1) \ll voShift$

En algunos casos, se puede aplicar un desplazamiento adicional durante la operación de desplazamiento del procedimiento de predicción descrito con respecto a la Tabla 6 anterior.

[0141] En la técnica de señalización de ejemplo de coeficientes de asignación de color para una función de asignación de color lineal para cada octante de la tabla de búsqueda 3D, pueden usarse ecuaciones similares para derivar o reconstruir los coeficientes de asignación de color (es decir, lutY, lutU, lutV) para la función de asignación de color lineal de la tabla de búsqueda 3D mediante la cuantización inversa de los valores residuales de los coeficientes de asignación de color, y la adición de los valores residuales cuantizados inversos de los coeficientes de asignación de color a los valores predichos de los coeficientes de asignación de color.

[0142] En algunos casos, los valores de los coeficientes de asignación de color lutY, lutU, lutV puede estar limitados a un cierto rango para reducir la complejidad computacional de la generación de la tabla de búsqueda 3D. Como ejemplo, se puede restringir el valor para que esté en el rango de -2^M a 2^M-1 , inclusive, donde M se establece igual a un valor fijo predefinido, tal como 10 o 12. De forma alternativa, el valor de M puede depender de uno o más del valor de bit de cuantización N , de la profundidad de bit de entrada (es decir, B) o de la profundidad de bit de salida (es decir, B_o) de la tabla de consulta 3D.

[0143] Después de que el codificador de vídeo 20 y/o el decodificador de vídeo 30 generen la tabla de búsqueda 3D usando una o más de las técnicas de ejemplo descritas anteriormente, la predicción de color se puede realizar de la siguiente manera usando la tabla de búsqueda 3D. La entrada al proceso de predicción de color es un triplete (y, u, v) en un espacio de color, por ejemplo, una primera gama de colores para una capa inferior o base de datos de vídeo. La salida del proceso de predicción de color es un triplete (Y, U, V) en otro espacio de color, por ejemplo, una segunda gama de colores para una capa superior o de mejora de datos de vídeo. Primero, el octante o cuboide más pequeño que cubre el triplete de entrada (y, u, v) se encuentra en la tabla de búsqueda 3D. Cada uno de los índices del vértice inicial del cuboide se derivan de la siguiente manera:

$$yIndex = y \gg (InputBitDepth - cur_octant_depth - cur_y_part_num_log2)$$

$$uIndex = u \gg (InputBitDepth - cur_octant_depth)$$

$$vIndex = v \gg (InputBitDepth - cur_octant_depth)$$

En algunos ejemplos, se puede aplicar un desplazamiento adicional durante el cálculo del índice. Entonces, otros tres índices del octante o cuboide se derivan como (yIndex, uIndex+1, vIndex); (yIndex, uIndex+1, vIndex+1); y (yIndex+1, uIndex+1, vIndex+1). Estos cuatro vértices pueden corresponder al cuarto caso de interpolación tetraédrica (P0, P1, P3, P7), que se ilustra como tetraedro 156D en la FIG. 12. El triplete de salida (Y, U, V) se obtiene entonces por interpolación tetraédrica, que interpola los valores de búsqueda 3D de los cuatro vértices. En otros ejemplos, se pueden usar otros casos de interpolación tetraédrica. De forma alternativa, pueden derivarse los ocho vértices del octante o cuboide. En este caso, se puede usar la interpolación trilineal para derivar el triplete de salida (Y, U, V).

[0144] En otro ejemplo, una tabla de búsqueda 3D puede señalizarse en el SPS o en el PPS. Entonces, en el encabezado de fragmento, se puede señalar un indicador adicional para indicar si la tabla de búsqueda 3D se sobrescribirá para el fragmento actual. De forma alternativa y/o adicionalmente, una tabla de búsqueda 3D puede señalizarse en el SPS y actualizarse en el PPS. Se debe tener en cuenta que la información común, tal como la profundidad máxima del octante, max_y_part_num_log2, input_bit_depth y output_bit_depth, solo se puede señalar en el nivel más alto, tal como en el SPS o en el PPS. Aquí max_octant_depth y max_y_part_num_log2 indican el número de partición máximo soportado de la tabla de búsqueda 3D. En algunos casos, dicha información puede estar relacionada con el perfil y/o el nivel en lugar de señalizarse al nivel más alto.

[0145] Como se describió anteriormente, en un ejemplo adicional, el codificador de vídeo 20 puede indicar condicionalmente un límite de partición para al menos uno de los componentes de croma (es decir, U o V) de una tabla de búsqueda 3D para el decodificador de vídeo 30 basándose en el al menos uno de los componentes de croma que se particiona en más de un octante, es decir, el valor de partición base es mayor que uno. En algunos casos, uno o ambos componentes de croma pueden no particionarse de manera uniforme. En otras palabras, para un componente de croma dado, al menos una de las particiones tiene un tamaño diferente al de las otras particiones. El límite de partición define una partición desigual de uno de los componentes de croma en dos o más octantes.

[0146] Convencionalmente, la información de límite de partición para cada componente de croma siempre se señala independientemente de si el componente de croma se particiona en dos o más segmentos u octantes. De acuerdo con las técnicas de esta divulgación, en un ejemplo, el codificador de vídeo 20 sólo señala el límite de partición cuando al menos uno de los componentes de croma (es decir, U o V) se particiona en dos o más partes. De lo contrario, el límite de partición es innecesario y no se señala al decodificador de vídeo. En otro ejemplo, el codificador de vídeo 20 sólo señala el límite de partición cuando cada uno de las componentes de croma (es decir, U y V) se particiona en dos o más partes.

[0147] En el ejemplo descrito con respecto a la Tabla 7 a continuación, la condición se basa en el elemento sintáctico `cur_octant_depth` que es igual a 1. Como se describió anteriormente con respecto a la Tabla 4 anterior, el elemento sintáctico `cur_octant_depth` indica un valor de partición base como la profundidad de división máxima para la tabla de búsqueda 3D. Cuando el elemento sintáctico `cur_octant_depth` es igual a 1, cada uno de los componentes de luma, del primer componente de croma y del segundo componente de croma se particionan en dos segmentos u octantes. En este caso, ambos componentes de croma (es decir, U y V) deben particionarse en dos partes para satisfacer la condición de señalización de información de límite de partición. Cualquier edición, adición o actualización del SHVC WD3, mencionada anteriormente, se indica con texto en *cursiva*.

TABLA 7: Sintaxis de la tabla de asignación de color

	Descriptor
<code>tabla de asignación de color() {</code>	
<code> cm_input_luma_bit_depth_minus8</code>	<code>u(3)</code>
<code> cm_input_chroma_bit_depth_delta</code>	<code>se(v)</code>
<code> cm_output_luma_bit_depth_minus8</code>	<code>u(3)</code>
<code> cm_output_chroma_bit_depth_delta</code>	<code>se(v)</code>
<code> si(<i>cur_octant_depth</i>==1) {</code>	
<code> <i>cb_part_threshold_minus_center</i></code>	<code>se(v)</code>
<code> <i>cr_part_threshold_minus_center</i></code>	<code>se(v)</code>
<code> }</code>	
<code> colour_mapping_matrix()</code>	
<code>}</code>	

El elemento sintáctico `cb_part_threshold_minus_center` especifica el límite de partición para el primer componente de croma (es decir, Cb). Cuando el elemento sintáctico `cb_part_threshold_minus_center` no está presente, se deduce como 0.

La variable `CbPartThreshold` se establece igual a $(1 \ll (\text{cm_input_luma_bit_depth_minus8} + \text{cm_input_chroma_bit_depth_delta} + 7)) + \text{cb_part_threshold_minus_center}$.

Cuando un valor de Cb es más pequeño o no mayor que la variable `CbPartThreshold`, el valor de Cb pertenece a la primera partición de Cb. De lo contrario, pertenece a la segunda partición de Cb. El elemento sintáctico `cr_part_threshold_minus_center` especifica el límite de partición para el segundo componente de croma (es decir, Cr). Cuando el elemento sintáctico `cr_part_threshold_minus_center` no está presente, se deduce como 0.

La variable `CrPartThreshold` se establece en $(1 \ll (\text{cm_input_luma_bit_depth_minus8} + \text{cm_input_chroma_bit_depth_delta} + 7)) + \text{cr_part_threshold_minus_center}$.

Cuando un valor de Cr es más pequeño, o no mayor, que la variable `CrPartThreshold`, el valor de Cr pertenece a la primera partición de Cr. De lo contrario, pertenece a la segunda partición Cr. Cabe señalar que los elementos sintácticos `cb_part_threshold_minus_center` y `cr_part_threshold_minus_center` no se cuantizan antes de la codificación.

[0148] La FIG. 14 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar técnicas para usar la tabla de búsqueda 3D basada en la escalabilidad de gama de colores en la codificación de vídeo multicapas. 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 de una imagen de vídeo determinada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El intramodo (modo I) puede referirse a cualquiera de varios modos de codificación de base espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la bipredicción (modo B), pueden referirse a cualquiera de varios modos de codificación de base temporal.

[0149] Como se muestra en la FIG. 14, el codificador de vídeo 20 recibe un bloque de vídeo actual dentro de una trama de vídeo que vaya a codificarse. En el ejemplo de la FIG. 14, el codificador de vídeo 20 incluye una unidad de selección de modo 40, una memoria de datos de vídeo 41, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformada 52, una unidad de cuantización 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modo 40 incluye una unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de procesamiento de intrapredicción 46, una unidad de partición

48 y una unidad de procesamiento de predicción de color 66. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye también una unidad de cuantización inversa 58, una unidad de procesamiento de transformada inversa 60 y un sumador 62. Puede incluirse también un filtro de desbloqueo (no mostrado en la FIG. 14) para filtrar fronteras de bloques, a fin de eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará típicamente la salida del sumador 62. También pueden usarse filtros adicionales (en bucle o posterior al 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).

[0150] La memoria de datos de vídeo 41 puede almacenar datos de vídeo que vayan a codificarse por los componentes del codificador de vídeo 20. Los datos de vídeo almacenados en la memoria de datos de vídeo 41 se pueden obtener, por ejemplo, a partir de la fuente de vídeo 18. La memoria intermedia de imágenes decodificadas 64 puede ser una memoria de imágenes de referencia que almacene datos de vídeo de referencia para su uso en la codificación de datos de vídeo mediante el codificador de vídeo 20, por ejemplo, en modos de intracodificación o de intercodificación. La memoria de datos de vídeo 41 y la memoria intermedia de imágenes decodificadas 64 pueden estar formadas por cualquiera entre varios dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 41 y la memoria intermedia de imágenes decodificadas 64 pueden proporcionarse mediante el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 41 puede estar en un chip con otros componentes del codificador de vídeo 20, o fuera de chip con respecto a esos componentes.

[0151] Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un fragmento de vídeo que vaya a codificarse. 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 realizan 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 procesamiento de intrapredicción 46, de forma alternativa, puede realizar la codificación intrapredictiva del bloque de vídeo recibido, con respecto a uno o más bloques contiguos de la misma trama o fragmento que el bloque que vaya a codificarse, para proporcionar predicción espacial. El codificador de vídeo 20 puede realizar múltiples pases de codificación, por ejemplo, para seleccionar una modalidad de codificación adecuada para cada bloque de datos de vídeo.

[0152] Además, la unidad de partición 48 puede particionar bloques de datos de vídeo en subbloques, basándose en la evaluación de los anteriores esquemas de división en los pases de codificación anteriores. Por ejemplo, la unidad de partición 48 puede particionar inicialmente una trama o un fragmento en varias LCU, y particionar cada una de las LCU en varias subCU, basándose en un análisis de velocidad-distorsión (por ejemplo, una optimización de velocidad-distorsión). La unidad de selección de modo 40 puede producir además una estructura de datos de árbol cuádruple, indicativa de la partición de una LCU en varias subCU. Las CU de nodos de hojas del árbol cuadrático pueden incluir una o más PU y una o más TU.

[0153] La unidad de selección de modo 40 puede seleccionar uno de los modos 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 bloque 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 intramodo, información de partición y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

[0154] 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 propósitos 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 de los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual, con respecto a un bloque predictivo dentro de una imagen de referencia (u otra unidad codificada) con respecto al bloque actual que se esté codificando dentro del fragmento actual (o a otra unidad codificada). Un bloque predictivo es un bloque que se revela como estrechamente coincidente con el bloque que se vaya a codificar, en lo que respecta a la diferencia de píxeles, lo cual puede determinarse mediante la suma de una diferencia absoluta (SAD), la suma de diferencia de cuadrados (SSD) u otras mediciones de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones de píxeles subenteros 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 de píxel fraccionario 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 completas y las posiciones de píxeles fraccionarias, y generar un vector de movimiento con una precisión de píxel fraccionaria.

[0155] 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.

5
 [0156] La compensación de movimiento, realizada 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 esté codificando, generando valores de diferencia de píxel, como se analiza a continuación. En general, la unidad de estimación de movimiento 42 realiza la estimación de movimiento con respecto a los componentes de luma, y la unidad de compensación de movimiento 44 usa 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 modo 40 también puede generar elementos sintácticos asociados con los bloques de vídeo y el fragmento de vídeo para su uso por el decodificador de vídeo 30 al decodificar los bloques de vídeo del fragmento de vídeo.

20
 [0157] La unidad de procesamiento de intrapredicción 46 puede intrapredecir un bloque actual, de forma alternativa a la intrapredicción realizada 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 procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción que se vaya a usar para codificar un bloque actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pases de codificación independientes, y la unidad de procesamiento de intrapredicción 46 (o la unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo adecuado de intrapredicción para su uso a partir de los modos probados.

30
 [0158] Por ejemplo, la unidad de procesamiento de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina, en general, una cantidad de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se ha codificado para generar el bloque codificado, así como una velocidad de bits (es decir, un número de bits) usada para generar el bloque codificado. La unidad de procesamiento de intrapredicción 46 puede calcular proporciones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intrapredicción presenta el mejor valor de velocidad-distorsión para el bloque.

40
 [0159] Después de seleccionar un modo de intrapredicción para un bloque, la unidad de intrapredicción 46 puede proporcionar información, indicativa del modo de intrapredicción seleccionado para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información indicando el modo de intrapredicción seleccionado. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido, que pueden incluir una pluralidad de tablas de índices de modos de intrapredicción y una pluralidad de tablas de índices de modos de intrapredicción modificadas (también denominadas tablas de asignación de contraseñas), definiciones de contextos de codificación para diversos bloques e indicaciones del modo de intrapredicción más probable, una tabla de índices de modos de intrapredicción y una tabla de índices de modos de intrapredicción modificadas para su uso para cada uno de los contextos.

50
 [0160] El codificador de vídeo 20 forma un bloque de vídeo residual restando los datos de predicción de la unidad de selección de modalidad 40 del bloque de vídeo original que esté codificándose. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformada 52 aplica una transformada, tal como una transformada discreta de coseno (DCT) o una transformada conceptualmente similar, al bloque residual, generando un bloque de vídeo que comprende valores residuales de coeficientes de transformada. La unidad de procesamiento de transformada 52 puede realizar otras transformadas que sean conceptualmente similares a la DCT. También podrían usarse transformadas de ondículas, transformadas de números enteros, transformadas de subbandas u otros tipos de transformadas. En cualquier caso, la unidad de procesamiento de transformadas 52 aplica la transformada al bloque residual, generando un bloque de coeficientes de transformada residuales. La transformada puede convertir la información residual, desde un dominio de valor de píxel en un dominio de transformada, tal como un dominio de frecuencia. La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantización 54.

60
 [0161] La unidad de cuantización 54 cuantiza los coeficientes de transformada para reducir más la tasa 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 entonces realizar un escaneo de la matriz que incluye los coeficientes de transformada cuantizados. De forma alternativa, la unidad de codificación por entropía 56 puede llevar a cabo el escaneo.

[0162] Tras la cuantización, la unidad de codificación por entropía 56 realiza la codificación por entropía de los coeficientes de transformada cuantizados. Por ejemplo, la unidad de codificación por entropía 56 puede realizar la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto y basada en sintaxis (SBAC), la codificación por entropía mediante la partición 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 archivar para su posterior transmisión o recuperación.

[0163] La unidad de cuantización inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican la cuantización inversa y la transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio del píxel, por ejemplo, para su posterior uso como 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 intermedia 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 subenteros para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque predictivo compensado por movimiento producido 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 usarse por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como bloque de referencia para intercodificar un bloque en una trama de vídeo subsiguiente.

[0164] De acuerdo con las técnicas descritas en esta divulgación, el codificador de vídeo 20 está configurado para realizar tablas de búsqueda 3D basadas en la escalabilidad de gama de colores cuando se codifiquen datos de vídeo multicapas. El codificador de vídeo 20 puede predecir y codificar datos de vídeo multicapas de acuerdo con cualquiera de la extensión SHVC, la extensión MV-HEVC y la extensión 3D-HEVC, u otras extensiones de codificación de vídeo multicapa. Específicamente, la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 puede generar imágenes de referencia intercapas usadas para predecir bloques de vídeo en una imagen de una capa superior de los datos de vídeo cuando una gama de colores para la capa superior de los datos de vídeo sea diferente de una gama de colores para una capa inferior de datos de vídeo.

[0165] La unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 puede realizar la predicción de color usando una tabla de búsqueda 3D para la escalabilidad de gama de colores para convertir datos de color de una imagen de referencia en una primera gama de colores para la capa inferior de los datos de vídeo en una segunda gama de colores para la capa superior de los datos de vídeo. En algunos ejemplos, la unidad de procesamiento de predicción de color 66 puede generar una tabla de búsqueda 3D independiente para cada uno de los componentes de color, es decir, un componente de luma, un primer componente de croma y un segundo componente de croma. Cada una de las tablas de búsqueda 3D incluye una dimensión de luma, una dimensión de primer croma y una dimensión de segundo croma, y se indexa usando los tres componentes de color independientes.

[0166] Las técnicas de la presente divulgación se refieren a la señalización de la información usada para generar tablas de búsqueda 3D para la escalabilidad de gama de colores. En algunos ejemplos de dichas técnicas, el codificador de vídeo 20 puede codificar información de partición y/o valores de color de una tabla de búsqueda 3D generada para la escalabilidad de gama de colores. Las técnicas descritas en esta divulgación pueden ser particularmente útiles para señalar la información usada para generar tablas de búsqueda 3D asimétricas y/o desequilibradas.

[0167] En un ejemplo de las técnicas divulgadas, la unidad de proceso de predicción de color 66 del codificador de vídeo 20 puede generar una tabla de búsqueda 3D con la partición gruesa para los primer y segundo componentes de croma y con la partición más fina para el componente de luma. La unidad de procesamiento de predicción de color 66 puede generar esta tabla de búsqueda 3D al particionar cada uno de los componentes de color de primer croma, de segundo croma y de luma en un primer número de octantes de acuerdo con un valor de partición base, por ejemplo, una profundidad de división máxima para la tabla de búsqueda 3D, y particionar entonces cada uno del primer número de octantes del componente de luma basándose en un valor de partición de luma. De esta manera, cada uno de los primer y segundo componentes de croma de la tabla de búsqueda 3D se particiona en un número menor o en menos octantes (es decir, partición más gruesa) y el componente de luma de la tabla de búsqueda 3D se particiona en un número mayor de o en más octantes (es decir, particiones más finas).

[0168] Como ejemplo, el valor de partición base es igual a 1 de manera que cada uno de los componentes de color se particiona en un único octante, y el valor de partición de luma es igual a 4, de manera que el octante único del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 4x1x1. Como otro ejemplo, el valor de partición base es igual a 2 de manera que cada uno de los componentes de color se particiona en dos octantes, y el valor de la partición de luma es igual a 4 de manera que cada uno de los dos octantes del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 8x2x2. Como puede verse, un valor de partición inferior da como resultado una partición más gruesa (es decir, un número menor de octantes) para un componente de color.

5 **[0169]** En algunos casos, la unidad de proceso de predicción de color 66 genera al menos un elemento sintáctico (por ejemplo, un primer elemento sintáctico) que indica el valor de partición de luma. En otros casos, el valor de partición de luma puede derivarse o conocerse tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. Como ejemplo, la unidad de procesamiento de predicción de color 66 puede derivar el valor de partición de luma basándose al menos en parte en el valor de partición base. En algunos casos, la unidad de procesamiento de predicción de color 66 también puede generar al menos un elemento sintáctico (por ejemplo, un segundo elemento sintáctico) que indique el valor de partición base. En otros casos, el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. La unidad de codificación por entropía 56 del codificador de vídeo 20 puede codificar por entropía entonces el primer y/o el segundo elemento sintáctico.

15 **[0170]** Además, el codificador de vídeo 20 puede codificar condicionalmente uno o más elementos sintácticos que indiquen un límite de partición para al menos uno de los componentes de croma. El límite de partición define una partición desigual de uno de los componentes de croma en dos o más octantes. De acuerdo con algunos ejemplos de las técnicas de esta divulgación, el codificador de vídeo 20 sólo codifica los elementos sintácticos que indican el límite de partición cuando al menos uno de los componentes de croma se particiona en más de un octante, es decir, cuando el valor de partición base es mayor que uno. De lo contrario, la señalización del límite de partición es innecesaria.

20 **[0171]** En otro ejemplo de las técnicas divulgadas, el codificador de vídeo 20 puede generar una tabla de búsqueda 3D basándose en una serie de octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma, y los valores de color para cada uno de los octantes. Como se describió anteriormente, en algunos casos, el codificador de vídeo 20 puede codificar al menos un elemento sintáctico que indique el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D. El codificador de vídeo 20 también puede codificar los valores de color para cada uno de los octantes para cada uno de los componentes de color. Por ejemplo, el codificador de vídeo 20 puede codificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D. La función de asignación de color lineal se usa para convertir datos de color en la primera gama de colores para la capa inferior de datos de vídeo en la segunda gama de colores para la capa superior de datos de vídeo. Los coeficientes de asignación de color para la función de asignación de color lineal son factores de ponderación entre los componentes de color de las capas inferiores y superiores de los datos de vídeo. Para cada uno de los componentes de color, uno de los coeficientes de asignación de color puede ser un coeficiente clave que defina un factor de ponderación entre el mismo componente de color de las capas inferior y más alta de los datos de vídeo.

35 **[0172]** Los coeficientes de asignación de color para la función de asignación de color lineal pueden derivarse como valores de punto flotante. Antes de codificar los coeficientes de asignación de color, la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 puede convertir los valores de punto flotante de los coeficientes de asignación de color en valores enteros. La conversión puede usar una profundidad de bits para los valores enteros basándose en al menos uno de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D. Además, la unidad de procesamiento de predicción de color 66 puede restringir los valores de los coeficientes de asignación de color para que estén dentro de un rango determinado basándose en un valor fijo predefinido o en un valor dependiente de al menos uno de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.

45 **[0173]** En algunos ejemplos de las técnicas de esta unidad de divulgación, el proceso de predicción de color 66 puede predecir uno o más de los coeficientes de asignación de color con el fin de codificar los valores residuales entre los valores originales de los coeficientes de asignación de color y los valores predichos de los coeficientes de asignación de color. Por ejemplo, para un primer octante para cada uno de los componentes de color, la unidad de procesamiento de predicción de color 66 puede predecir los coeficientes de asignación de color de la función de asignación de color lineal basándose en valores fijos predefinidos. En un ejemplo, para un primer octante para cada uno de los componentes de color, la unidad de procesamiento de predicción de color 66 puede codificar un coeficiente clave de la función de asignación de color lineal basándose en un valor predicho igual a un valor predefinido distinto de cero, y codificar cualquier coeficiente de asignación de color restante de la función de asignación de color lineal basándose en un valor predicho igual a cero. En este ejemplo, la unidad de procesamiento de predicción de color 66 puede codificar los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color basándose en valores predichos de al menos un octante codificado previamente, tal como el primer octante.

60 **[0174]** La unidad de codificación por entropía 56 del codificador de vídeo 20 puede codificar por entropía entonces los valores residuales de los coeficientes de asignación de color para la función de asignación de color lineal para cada uno de los octantes para cada uno de los componentes de color. En algunos casos, antes de la codificación por entropía, el codificador de vídeo 20 puede cuantizar los valores residuales de los coeficientes de asignación de color usando la unidad de cuantización 54 basándose en un valor de cuantización determinado. El codificador de vídeo 20 puede codificar el valor de cuantización determinado.

65 **[0175]** Al generar la tabla de búsqueda 3D, la unidad de procesamiento de predicción de color 66 realiza la predicción de color de una imagen de referencia para la capa inferior de los datos de vídeo usando la tabla de búsqueda 3D, y genera al menos una imagen de referencia intercapas para la capa superior de los datos de vídeo basándose en la

imagen de referencia de color predicho. Al generar la imagen de referencia intercapas, la unidad de compensación de movimiento 44 del codificador de vídeo 20 puede funcionar como se describe anteriormente para predecir bloques de vídeo en una imagen de la capa superior de los datos de vídeo basándose en las imágenes de referencia intercapas generadas usando la tabla de búsqueda 3D. El codificador de vídeo 20 puede codificar entonces datos residuales de los bloques de vídeo en un flujo de bits para su transmisión al decodificador de vídeo 30.

[0176] La FIG. 15 es un diagrama de bloques que ilustra un ejemplo de decodificador de vídeo 30 que puede implementar técnicas para determinar el uso de la tabla de búsqueda 3D basada en la escalabilidad de gama de colores en la codificación de vídeo multicapas. En el ejemplo de la FIG. 15, el decodificador de vídeo 30 incluye una unidad de decodificación por entropía 70, una memoria de datos de vídeo 71, una unidad de compensación de movimiento 72, una unidad de procesamiento de intrapredicción 74, una unidad de procesamiento de predicción de color 86, una unidad de cuantización inversa 76, una unidad de procesamiento de transformada inversa 78, una memoria de imágenes de referencia 82 y un sumador 80. En algunos ejemplos, el decodificador de vídeo 30 puede realizar un pase de decodificación en general recíproco al pase de codificación descrito con respecto al codificador de vídeo 20 (FIG. 14). 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.

[0177] La memoria de datos de vídeo 71 puede almacenar datos de vídeo, tales como un flujo de bits de vídeo codificado, para decodificarse por los componentes del decodificador de vídeo 30. Los datos de vídeo almacenados en la memoria de datos de vídeo 71 pueden obtenerse, por ejemplo, a partir del medio legible por ordenador 16, por ejemplo, desde una fuente de vídeo local, tal como una cámara, mediante comunicación de datos de vídeo por red cableada o inalámbrica, o accediendo a medios de almacenamiento físico de datos. La memoria de datos de vídeo 71 puede formar una memoria intermedia de imágenes codificadas (CPB) que almacene datos de vídeo codificados a partir de un flujo de bits de vídeo codificado. La memoria intermedia de imágenes decodificadas 82 puede ser una memoria de imágenes de referencia que almacene datos de vídeo de referencia para su uso en la decodificación de datos de vídeo mediante el decodificador de vídeo 30, por ejemplo, en modos de intracodificación o de intercodificación. La memoria de datos de vídeo 71 y la memoria intermedia de imágenes decodificadas 82 pueden estar formadas por cualquiera de entre varios dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 71 y la memoria intermedia de imágenes decodificadas 82 pueden proporcionarse mediante el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 71 puede estar en un chip con otros componentes del decodificador de vídeo 30, o fuera de chip con respecto a esos componentes.

[0178] Durante el proceso de decodificación, el decodificador de vídeo 30 recibe un flujo de bits de vídeo codificado que representa los bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados del 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 en el nivel del bloque de vídeo.

[0179] Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), la unidad de procesamiento de intrapredicción 74 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual, basándose en un modo de intrapredicción señalado, y datos de bloques previamente decodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (es decir, B o P), 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 imágenes de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por defecto basándose en las 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 los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos del bloque de vídeo actual que se esté decodificando. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción), usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B o fragmento P), 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.

[0180] 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 usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques predictivos.

[0181] La unidad de cuantización inversa 76 cuantifica de manera inversa, es decir, descuantiza, los coeficientes de transformada 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 QP_Y , 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. La unidad de procesamiento de transformada inversa 78 aplica una transformada inversa, por ejemplo una DCT inversa, una transformada inversa entera o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada, con el fin de generar bloques residuales en el dominio de píxeles.

[0182] 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 en otros elementos sintácticos, el decodificador de vídeo 30 forma un bloque de vídeo decodificado sumando los bloques residuales procedentes de la unidad de procesamiento de transformada inversa 78 a 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 eliminación de bloques para filtrar los bloques decodificados con el fin de eliminar distorsiones de efecto pixelado. También pueden usarse 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 de 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 subsiguiente compensación de movimiento. La memoria intermedia 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 FIG. 1.

[0183] De acuerdo con algunos ejemplos de las técnicas descritas en esta divulgación, el decodificador de vídeo 30 está configurado para realizar la tabla de búsqueda 3D basada en la escalabilidad de gama de colores cuando se decodifiquen datos de vídeo multicapas. El decodificador de vídeo 30 puede decodificar y reconstruir los datos de vídeo multicapa predichos de acuerdo con cualquiera de la extensión SHVC, de la extensión MV-HEVC, de la extensión 3D-HEVC o de otras extensiones de codificación de vídeo multicapas a HEVC. Específicamente, la unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 puede generar imágenes de referencia intercapas usadas para predecir bloques de vídeo en una imagen de una capa superior de los datos de vídeo cuando una gama de colores para la capa superior de los datos de vídeo sea diferente a una gama de colores para una capa inferior de datos de vídeo.

[0184] La unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 puede realizar la predicción de color usando una tabla de búsqueda 3D para la escalabilidad de gama de colores para convertir los datos de color de una imagen de referencia en una primera gama de colores para la capa inferior de los datos de vídeo en una segunda gama de colores para la capa superior de los datos de vídeo. En algunos ejemplos, la unidad de procesamiento de predicción de color 86 puede generar una tabla de búsqueda 3D independiente para cada uno de los componentes de color, es decir, un componente de luma, un primer componente de croma y un segundo componente de croma. Cada una de las tablas de búsqueda 3D incluye una dimensión de luma, una dimensión de primer croma y una dimensión de segundo croma, y se indexa usando los tres componentes de color independientes.

[0185] Las técnicas de la presente divulgación se refieren a la señalización de la información usada para generar tablas de búsqueda 3D para la escalabilidad de gama de colores. De acuerdo con las técnicas, el decodificador de vídeo 30 puede decodificar la información de partición y/o los valores de color para generar una tabla de búsqueda 3D con el fin de realizar la escalabilidad de gama de colores. Las técnicas descritas en esta divulgación pueden ser particularmente útiles para señalar la información usada para generar tablas de búsqueda 3D asimétricas y/o desequilibradas.

[0186] En un ejemplo de las técnicas descritas, la unidad de proceso de predicción de color 86 del decodificador de vídeo 30 puede generar una tabla de búsqueda 3D con la partición gruesa para los primer y segundo componentes de croma y con la partición más fina para el componente de luma. La unidad de procesamiento de predicción de color 86 puede generar esta tabla de búsqueda 3D al particionar cada uno de los componentes de color de primer croma, de segundo croma y de luma en un primer número de octantes de acuerdo con un valor de partición base, por ejemplo, una profundidad de división máxima para la tabla de búsqueda 3D, y particionar entonces cada uno del primer número de octantes del componente de luma basado en un valor de partición de luma. De esta manera, cada uno de los componentes de primer y segundo croma de la tabla de búsqueda 3D se particiona en un número menor o en menos octantes (es decir, partición más gruesa) y el componente de luma de la tabla de búsqueda 3D se particiona en un número mayor de o en más octantes (es decir, particiones más finas).

- 5 [0187] Como ejemplo, el valor de partición base es igual a 1 de manera que cada uno de los componentes de color se particiona en un único octante, y el valor de partición de luma es igual a 4, de manera que el octante único del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 4x1x1. Como otro ejemplo, el valor de partición base es igual a 2 de manera que cada uno de los componentes de color se particiona en dos octantes y el valor de la partición de luma es igual a 4 de manera que cada uno de los dos octantes del componente de luma se particiona en cuatro octantes, lo que da como resultado una tabla de búsqueda 3D de tamaño 8x2x2. Como puede verse, un valor de partición inferior da como resultado una partición más gruesa (es decir, un número menor de octantes) para un componente de color.
- 10 [0188] En algunos casos, la unidad de decodificación por entropía 70 del decodificador de vídeo 30 decodifica por entropía al menos un elemento sintáctico (por ejemplo, un primer elemento sintáctico) que indica el valor de partición de luma. En otros casos, el valor de partición de luma puede derivarse o conocerse tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. Como ejemplo, la unidad de procesamiento de predicción de color 86 puede derivar el valor de partición de luma basándose al menos en parte en el valor de partición base. En algunos casos, la
- 15 unidad de decodificación por entropía 70 también puede decodificar al menos un elemento sintáctico (por ejemplo, un segundo elemento sintáctico) que indique el valor de partición base. En otros casos, el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. La unidad de procesamiento de predicción de color 86 usa el valor de partición base predefinido o señalado y el valor de partición de luma derivado o señalado para generar la tabla de búsqueda 3D con una partición más gruesa para los
- 20 primer y segundo componentes de croma y con una partición más fina para el componente de luma, como se describió anteriormente.
- [0189] Además, el decodificador de vídeo 30 puede decodificar de forma condicional uno o más elementos sintácticos que indiquen un límite de partición para al menos uno de los componentes de croma. El límite de partición define una
- 25 partición desigual de uno de los componentes de croma en dos o más octantes. De acuerdo con las técnicas de esta divulgación, el decodificador de vídeo 30 sólo decodifica los elementos sintácticos que indican el límite de partición cuando al menos uno de los componentes de croma se particiona en más de un octante, es decir, cuando el valor de partición base es mayor que uno. De lo contrario, decodificar el límite de partición es innecesario.
- 30 [0190] En otro ejemplo de las técnicas descritas, el decodificador de vídeo 30 puede generar una tabla de búsqueda 3D basándose en una serie de octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma, y en los valores de color para cada uno de los octantes. Como se describió anteriormente, en algunos casos, el decodificador de vídeo 30 puede decodificar al menos un elemento sintáctico que indique el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D, o bien determinar el número de
- 35 octantes para cada uno de los componentes de color de la tabla de búsqueda 3D. El decodificador de vídeo 30 también puede decodificar los valores de color para cada uno de los octantes para cada uno de los componentes de color. Por ejemplo, el decodificador de vídeo 30 puede decodificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D. La función de asignación de color lineal se usa para convertir datos de color en la primera gama de colores para la capa inferior de datos de vídeo en la
- 40 segunda gama de colores para la capa superior de datos de vídeo. Los coeficientes de asignación de color para la función de asignación de color lineal son factores de ponderación entre los componentes de color de las capas inferiores y superiores de los datos de vídeo. Para cada uno de los componentes de color, uno de los coeficientes de asignación de color puede ser un coeficiente clave que defina un factor de ponderación entre el mismo componente de color de las capas inferior y superior de los datos de vídeo.
- 45 [0191] Los coeficientes de asignación de color para la función de asignación de color lineal se derivan primero como valores de punto flotante. Los valores de punto flotante se convierten o se cuantizan a valores enteros y se señalizan como valores enteros. La conversión puede usar una profundidad de bits para los valores enteros basándose en al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.
- 50 Además, la unidad de procesamiento de predicción de color 86 puede restringir los valores de los coeficientes de asignación de color para que estén dentro de un rango determinado basándose en un valor fijo predefinido o en un valor dependiente de al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.
- 55 [0192] La unidad de decodificación por entropía 70 del decodificador de vídeo 30 puede decodificar por entropía valores residuales de decodificación de los coeficientes de asignación de color para la función de asignación de color lineal para cada uno de los octantes para cada uno de los componentes de color. En algunos casos, después de la decodificación por entropía y antes de la reconstrucción, el decodificador de vídeo 30 puede cuantizar a la inversa los valores residuales de los coeficientes de asignación de color usando la unidad de cuantización inversa 76 basándose
- 60 en un valor de cuantización determinado. El decodificador de vídeo 30 puede decodificar un elemento sintáctico que indique el valor de cuantización determinado.
- [0193] De acuerdo con las técnicas de esta divulgación, la unidad de procesamiento de predicción de color 86 puede predecir uno o más de los coeficientes de asignación de color para reconstruir los valores de los coeficientes de
- 65 asignación de color basados en los valores residuales de los coeficientes de asignación de color y en los valores predichos de los coeficientes de asignación de color. Por ejemplo, para un primer octante para cada uno de los

componentes de color, la unidad de procesamiento de predicción de color 86 puede predecir los coeficientes de asignación de color de la función de asignación de color lineal basándose en valores fijos predefinidos. En un ejemplo, para un primer octante para cada uno de los componentes de color, la unidad de procesamiento de predicción de color 86 puede decodificar un coeficiente clave de la función de asignación de color lineal basándose en un valor predicho igual a un valor distinto de cero predefinido, y decodificar cualquier coeficiente de asignación de color restante de la función de asignación de color lineal basándose en un valor predicho igual a cero. En este ejemplo, la unidad de procesamiento de predicción de color 86 puede decodificar los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color basándose en valores predichos de al menos un octante decodificado previamente, tal como el primer octante.

[0194] Tras generar la tabla de búsqueda 3D, la unidad de procesamiento de predicción de color 86 realiza la predicción del color de una imagen de referencia para la capa inferior de los datos de vídeo usando la tabla de búsqueda 3D, y genera una imagen de referencia intercapas para la capa superior de los datos de vídeo basándose en la imagen de referencia de color predicho. Tras generar las imágenes de referencia intercapas, la unidad de compensación de movimiento 72 del decodificador de vídeo 30 puede funcionar como se describió anteriormente para reconstruir bloques de vídeo en una imagen de la capa superior de los datos de vídeo basándose en datos residuales decodificados y en las imágenes de referencia intercapas generadas usando la tabla de búsqueda 3D.

[0195] La FIG. 16 es un diagrama de flujo que ilustra una operación de ejemplo de codificación de información de partición para al menos uno de los componentes de color de una tabla de búsqueda 3D. La operación de ejemplo de la FIG. 16 se describe en el presente documento como realizado por la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 de la FIG. 14. En otros ejemplos, la operación puede realizarse mediante la unidad de procesamiento de predicción de color 144 de la FIG. 8.

[0196] De acuerdo con las técnicas de esta divulgación, la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 puede generar una tabla de búsqueda 3D con particiones más gruesas para los componentes de primer y segundo croma y con una partición más fina para el componente de luma. La unidad de procesamiento de predicción de color 66 puede generar esta tabla de búsqueda 3D al particionar cada uno de los componentes de luma, de primer croma y de segundo croma de la tabla de búsqueda 3D en un primer número de octantes basándose en un valor de partición base (180). En un ejemplo, el valor de partición base puede ser una profundidad de división máxima para la tabla de búsqueda 3D. La unidad de procesamiento de predicción de color 66 particiona entonces además cada uno de los primeros octantes numéricos del componente de luma en un segundo número de octantes basándose en un valor de partición de luma (182).

[0197] En algunos casos, el codificador de vídeo 20 puede codificar al menos un elemento sintáctico (por ejemplo, un primer elemento sintáctico) que indique el valor de partición de luma para el componente de luma de la tabla de búsqueda 3D. En otros casos, el valor de partición de luma puede derivarse o conocerse tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. En algunos casos, el codificador de vídeo 20 también puede generar al menos un elemento sintáctico adicional (por ejemplo, un segundo elemento sintáctico) que indique el valor de partición base para la tabla de búsqueda 3D. En otros casos, el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30.

[0198] Además, el codificador de vídeo 20 puede codificar condicionalmente uno o más elementos sintácticos que indiquen un límite de partición para al menos uno de los componentes de croma. El límite de partición define una partición desigual de uno de los componentes de croma en dos o más octantes. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 codifica los elementos sintácticos que indican el límite de partición para al menos uno de los componentes de croma basándose en uno de los componentes de croma que se particiona en más de un octante, es decir, el valor de partición base mayor que uno.

[0199] El codificador de vídeo 20 también puede codificar valores de color para cada uno de los octantes para cada uno de los componentes de color. Por ejemplo, el codificador de vídeo 20 puede codificar valores de color de vértices para cada uno de los octantes de cada uno de los componentes de color. Como otro ejemplo, el codificador de vídeo 20 puede codificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D. De esta manera, un decodificador de vídeo, tal como el decodificador de vídeo 30 de la FIG. 15, puede generar una tabla de búsqueda 3D basándose en la información de partición señalizada y en los valores de color señalizados para realizar una escalabilidad de gama de colores para decodificar datos de vídeo multicapas.

[0200] La FIG. 17 es un diagrama de flujo que ilustra una operación de ejemplo de decodificación de información de partición para al menos uno de los componentes de color de una tabla de búsqueda 3D. La operación de ejemplo de la FIG. 17 se describe en el presente documento como realizado por la unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 de la FIG. 15. En otros ejemplos, la operación puede realizarse mediante la unidad de procesamiento de predicción de color 144 de la FIG. 8.

[0201] De acuerdo con las técnicas de esta divulgación, el decodificador de vídeo 30 determina un valor de partición base para la tabla de búsqueda 3D (186). En algunos casos, el decodificador de vídeo 30 puede decodificar, desde un flujo de bits recibido, al menos un elemento sintáctico (por ejemplo, un segundo elemento sintáctico) que indique el valor de la partición base. En otros casos, el valor de partición base puede ser un valor predefinido conocido tanto en el codificador de vídeo 20 como en el decodificador de vídeo 30. El decodificador de vídeo 30 también determina un valor de partición de luma para un componente de luma de la tabla de búsqueda 3D (188). En algunos casos, el decodificador de vídeo 30 puede decodificar, desde el flujo de bits recibido, al menos un elemento sintáctico (por ejemplo, un primer elemento sintáctico) que indica el valor de partición de luma. En otros casos, el decodificador de vídeo 30 puede derivar el valor de partición de luma. En un ejemplo, el decodificador de vídeo 30 puede derivar el valor de partición de luma basándose al menos en parte en el valor de partición base.

[0202] La unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 usa el valor de partición base y el valor de partición luma para generar la tabla de búsqueda 3D con particiones más gruesas para los primer y segundo componentes de croma y con una partición más fina para el componente de luma. La unidad de procesamiento de predicción de color 86 puede generar esta tabla de búsqueda 3D al particionar cada uno de los componentes de luma, de primer croma y de segundo croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base (190). En un ejemplo, el valor de partición base puede ser una profundidad de división máxima para la tabla de búsqueda 3D. La unidad de procesamiento de predicción de color 86 particiona entonces cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en un valor de partición de luminancia (192). De esta manera, el componente de luma se puede particionar para tener un número mayor de octantes que cada uno de los componentes de croma.

[0203] Además, el decodificador de vídeo 30 puede decodificar de forma condicional uno o más elementos sintácticos que indiquen un límite de partición para al menos uno de los componentes de croma. El límite de partición define una partición desigual de uno de los componentes de croma en dos o más octantes. De acuerdo con las técnicas de esta divulgación, el decodificador de vídeo 30 decodifica los elementos sintácticos que indican el límite de partición para al menos uno de los componentes de croma basándose en el componente de croma que se particiona en más de un octante, es decir, el valor de partición base mayor que uno.

[0204] El decodificador de vídeo 30 también puede decodificar valores de color para cada uno de los octantes para cada uno de los componentes de color. Por ejemplo, el decodificador de vídeo 30 puede decodificar valores de color de vértices para cada uno de los octantes de cada uno de los componentes de color. Como otro ejemplo, el decodificador de vídeo 30 puede decodificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D. De esta manera, el decodificador de vídeo 30 puede generar una tabla de búsqueda 3D basándose en la información de partición señalizada y en los valores de color señalizados para realizar la escalabilidad de gama de colores para decodificar datos de vídeo multicapas.

[0205] La FIG. 18 es un diagrama de flujo que ilustra una operación de ejemplo de codificación de valores de color para cada uno de los octantes para cada uno de los componentes de color de una tabla de búsqueda 3D. La operación de ejemplo de la FIG. 18 se describe en el presente documento como realizado por la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 de la FIG. 14. En otros ejemplos, la operación puede realizarse mediante la unidad de procesamiento de predicción de color 144 de la FIG. 8.

[0206] De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede generar una tabla de búsqueda 3D basándose en varios octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma, y valores de color para cada uno de los octantes (200). El codificador de vídeo 20 puede codificar los valores de color para cada uno de los octantes para cada uno de los componentes de color. Más específicamente, para cada uno de los octantes para cada uno de los componentes de color, el codificador de vídeo 20 puede codificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D (202).

[0207] Antes de codificar los coeficientes de asignación de color, la unidad de procesamiento de predicción de color 66 del codificador de vídeo 20 puede convertir los valores de punto flotante de los coeficientes de asignación de color en valores enteros usando una profundidad de bits basada en al menos una de una entrada de profundidad de bits o de una profundidad de bits de salida de la tabla de búsqueda 3D. Además, la unidad de procesamiento de predicción de color 66 puede restringir los valores de los coeficientes de asignación de color para que estén dentro de un rango determinado basándose en un valor fijo predefinido o en un valor dependiente de al menos uno de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.

[0208] La unidad de procesamiento de predicción de color 66 puede predecir uno o más de los coeficientes de asignación de color con el fin de codificar valores residuales entre los valores originales de los coeficientes de asignación de color y los valores predichos de los coeficientes de asignación de color. Por ejemplo, para un primer octante para cada uno de los componentes de color, la unidad de procesamiento de predicción de color 66 puede codificar un coeficiente clave de la función de asignación de color lineal basándose en un valor predicho igual a un valor predefinido distinto de cero, y codificar cualquier coeficiente de asignación de color restante de la función de asignación de color lineal basándose en un valor predicho igual a cero. En este ejemplo, la unidad de procesamiento

de predicción de color 66 puede codificar los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color basándose en valores predichos de al menos un octante codificado previamente, tal como el primer octante. En algunos casos, antes de codificar los valores residuales de los coeficientes de asignación de color, el codificador de vídeo 20 pueden cuantizar los valores residuales de los coeficientes de asignación de color basándose en un valor de cuantización determinado.

[0209] El codificador de vídeo 20 también puede codificar al menos un elemento sintáctico que indique el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D. De esta manera, un decodificador de vídeo, tal como el decodificador de vídeo 30 de la FIG. 15, puede generar una tabla de búsqueda 3D basándose en la información de partición señalizada y en los valores de color señalizados para realizar la escalabilidad de gama de colores para decodificar datos de vídeo multicapas.

[0210] La FIG. 19 es un diagrama de flujo que ilustra una operación de ejemplo de decodificación de valores de color para cada uno de los octantes para cada uno de los componentes de color de una tabla de búsqueda 3D. La operación de ejemplo de la FIG. 19 se describe en el presente documento como realizado por la unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 de la FIG. 15. En otros ejemplos, la operación puede realizarse mediante la unidad de procesamiento de predicción de color 144 de la FIG. 8.

[0211] De acuerdo con algunos ejemplos de las técnicas de esta divulgación, el decodificador de vídeo 30 determina varios octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma de la tabla de búsqueda 3D (204). En algunos casos, el decodificador de vídeo 30 puede decodificar, a partir de un flujo de bits recibido, al menos un elemento sintáctico que indique el número de octantes para al menos uno de los componentes de color de la tabla de búsqueda 3D. El decodificador de vídeo 30 también decodifica los valores de color para cada uno de los octantes para cada uno de los componentes de color. Más específicamente, para cada uno de los octantes para cada uno de los componentes de color, el decodificador de vídeo 30 puede decodificar los coeficientes de asignación de color para una función de asignación de color lineal de los valores de color en la tabla de búsqueda 3D (206). La unidad de procesamiento de predicción de color 86 del decodificador de vídeo 30 genera entonces la tabla de búsqueda 3D basándose en el número de octantes para cada uno de los componentes de color de luma, de primer croma y de segundo croma, y los valores de color asociados con los coeficientes de asignación de color para cada uno de los octantes (208). El decodificador de vídeo 30 puede usar la tabla de búsqueda 3D para realizar escalabilidad de gama de colores para decodificar datos de vídeo multicapas.

[0212] El decodificador de vídeo 30 puede recibir valores residuales de los coeficientes de asignación de color para la función de asignación de color lineal para cada uno de los octantes para cada uno de los componentes de color. En algunos casos, después de decodificar los valores residuales de los coeficientes de asignación de color, el decodificador de vídeo 30 puede cuantizar a la inversa los valores residuales de los coeficientes de asignación de color basándose en un valor de cuantización determinado. La unidad de procesamiento de predicción de color 86 puede predecir entonces uno o más de los coeficientes de asignación de color para reconstruir los valores de los coeficientes de asignación de color basándose en los valores residuales señalados de los coeficientes de asignación de color y en los valores predichos de los coeficientes de asignación de color. Por ejemplo, para un primer octante para cada uno de los componentes de color, la unidad de procesamiento de predicción de color 86 puede decodificar un coeficiente clave de la función de asignación de color lineal basándose en un valor predicho igual a un valor distinto de cero predefinido, y decodificar cualquier coeficiente de asignación de color restante de la función de asignación de color lineal basándose en un valor predicho igual a cero. En este ejemplo, la unidad de procesamiento de predicción de color 86 puede decodificar los coeficientes de asignación de color de cualquier octante restante para cada uno de los componentes de color basándose en valores predichos de al menos un octante decodificado previamente, tal como el primer octante.

[0213] Después de descodificar los coeficientes de asignación de color, los coeficientes de asignación de color pueden ser valores enteros que representen valores de punto flotante usando una profundidad de bits basándose al menos en uno de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D. La unidad de procesamiento de predicción de color 86 puede restringir los valores de los coeficientes de asignación de color para que estén dentro de un rango determinado basándose en un valor fijo predefinido o en un valor dependiente de al menos una de una profundidad de bits de entrada o de una profundidad de bits de salida de la tabla de búsqueda 3D.

[0214] Ciertos aspectos de esta divulgación se han descrito con respecto a las ampliaciones de la norma de HEVC, para los propósitos de ilustración. Sin embargo, las técnicas descritas en esta divulgación pueden ser útiles para otros procesos de codificación de vídeo, que incluyen otros procesos de codificación de vídeo, estándar o patentados, aún no desarrollados.

[0215] Un codificador de vídeo, tal como se describe en esta divulgación, puede referirse a un codificador de vídeo o a un decodificador de vídeo. De manera similar, una unidad de codificación de vídeo puede referirse a un codificador de vídeo o a un decodificador de vídeo. Del mismo modo, la codificación de vídeo puede referirse a la codificación de vídeo o a la decodificación de vídeo, según corresponda.

[0216] Debe reconocerse que, dependiendo del ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden agregarse, fusionarse 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.

[0217] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o en cualquier combinación de los mismos. Si se implementa en el software, las funciones pueden almacenarse o transmitirse como una o más instrucciones o código en un medio legible por ordenador y ejecutarse por 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 tales como los 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 sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0218] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender memoria 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 utilizarse para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe debidamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una página web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, están dirigidos 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, donde algunos discos habitualmente emiten datos magnéticamente, mientras que otros discos emiten datos ópticamente con láseres. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

[0219] Las instrucciones pueden ejecutarse por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de uso general, circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA) u otra circuitería lógica equivalente, integrada o discreta. En consecuencia, el término «procesador», como se usa en el presente documento, puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de hardware y/o software dedicados, configurados para la codificación y la decodificación, o incorporados en un códec combinado. Asimismo, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.

[0220] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, que incluyen un teléfono inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante 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, incluyendo uno o más procesadores, como se ha descrito anteriormente, conjuntamente con software y/o firmware adecuados.

[0221] Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento de decodificación de datos de vídeo, el procedimiento que comprende:
 - 5 determinación de un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores;
 - determinar un valor de partición de luma para un componente de luma de la tabla de búsqueda 3D;
 - 10 generar la tabla de búsqueda 3D con particiones más gruesas para los componentes de croma y con particiones más finas para el componente de luma, que incluyen:
 - 15 particionar cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en el valor de partición base, y
 - particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en el valor de partición de luma de manera que un número total de octantes para el componente de luma sea mayor que un número total de octantes para el primer componente de croma y mayor que un número total de octantes para el segundo componente de croma;
 - 20 decodificar datos residuales de bloques de vídeo de los datos de vídeo; y
 - reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos una imagen de referencia generada usando la tabla de búsqueda 3D.
 - 25 2. El procedimiento según la reivindicación 1, en el que determinar el valor de partición de luma comprende decodificar al menos un elemento sintáctico que indica el valor de partición de luma para el componente de luma de la tabla de búsqueda 3D.
 - 30 3. El procedimiento según la reivindicación 1, en el que determinar el valor de partición de luma comprende derivar el valor de partición de luma para el componente de luma de la tabla de búsqueda 3D.
 4. El procedimiento según la reivindicación 1, en el que determinar el valor de partición base comprende decodificar al menos un elemento sintáctico que indique el valor de partición base para el componente de luma, el primer componente de croma y el segundo componente de croma de la tabla de búsqueda 3D.
 - 35 5. El procedimiento según la reivindicación 1, en el que determinar el valor de partición base comprende determinar el valor de partición base como una profundidad de división predefinida para el componente de luma, el primer componente de croma y el segundo componente de croma de la tabla de búsqueda 3D.
 - 40 6. El procedimiento según la reivindicación 1, en el que la partición más gruesa para los componentes de croma comprende un número menor de octantes en cada uno del primer componente de croma y del segundo componente de croma que en el componente de luma, y la partición más fina para el componente de luma comprende un número mayor de octante en el componente de luma que en cada uno del primer componente de croma y del segundo componente de croma.
 - 45 7. El procedimiento según la reivindicación 1, que comprende además, basándose en al menos uno del primer componente de croma o del segundo componente de croma que se particiona en más de un octante, que decodifica uno o más elementos sintácticos que indican un límite de partición para al menos uno del primer componente de croma o del segundo componente de croma, en el que el límite de partición define una partición desigual del al menos uno del primer componente de croma o el segundo componente de croma en el primer número de octantes.
 - 50 8. El procedimiento según la reivindicación 1, en el que la generación de la tabla de búsqueda en 3D incluye además determinar los valores de color para cada uno de los octantes de cada uno de los componentes de luma, del primer componente de croma y del segundo componente de croma.
 - 55 9. El procedimiento según la reivindicación 8, en el que los valores de color comprenden valores de vértices de cada uno de los octantes, o valores de coeficientes de asignación de color de una función de asignación de color lineal para cada uno de los octantes.
 - 60 10. El procedimiento según la reivindicación 1, que comprende además:
 - 65 realizar la predicción de color usando la tabla de búsqueda 3D para convertir los datos de color de una imagen de referencia en una primera gama de colores para una capa inferior de los datos de vídeo en una segunda gama de colores para una capa superior de los datos de vídeo; y

generar al menos una imagen de referencia intercapas para la capa superior de los datos de vídeo basándose en los datos de color convertidos,

5 en el que la reconstrucción de los bloques de vídeo de los datos de vídeo comprende la reconstrucción de bloques de vídeo en una imagen de la capa superior de los datos de vídeo basándose en los datos residuales decodificados y la al menos una imagen de referencia intercapas generada usando la tabla de búsqueda 3D.

10 11. Un procedimiento de codificación de datos de vídeo, el procedimiento que comprende:

generar una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores con particiones más gruesas para los componentes de croma y con particiones más finas para un componente de luma, que incluyen:

15 particionar cada uno de los componentes de luma, de un primer componente de croma y un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en un valor de partición base para la tabla de búsqueda 3D, y

20 particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en un valor de partición de luma para el componente de luma de la tabla de búsqueda 3D, de tal manera que un número total de octantes para el componente de luma sea mayor que un número total de octantes para el primer componente de croma y mayor que un número total de octantes para el segundo componente de croma;

25 predecir bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D; y

30 codificar datos residuales de los bloques de vídeo en un flujo de bits.

12. El procedimiento según la reivindicación 11, que comprende además codificar al menos un elemento sintáctico que indique el valor de la partición de luma para el componente de luma de la tabla de búsqueda 3D.

35 13. Un dispositivo de decodificación de vídeo que comprende:

una memoria configurada para almacenar datos de vídeo; y

uno o más procesadores en comunicación con la memoria, y configurados para:

40 determinar un valor de partición base para una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo,

determinar un valor de partición luma para un componente de luma de la tabla de búsqueda 3D,

45 generar la tabla de búsqueda 3D con particiones más gruesas para los componentes de croma y con una partición más fina para el componente de luma, el uno o más procesadores configurados para particionar cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda de 3D en un primer número de octantes basándose en el valor de la partición base y particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en el valor de la partición de luma de manera que el número total de octantes para el componente de luma sea mayor que un número total de octantes para el primer componente de croma y mayor que un número total de octantes para el segundo componente de croma,

55 decodificar datos residuales de bloques de vídeo de los datos de vídeo, y

reconstruir los bloques de vídeo de los datos de vídeo basándose en los datos residuales decodificados y al menos una imagen de referencia generada usando la tabla de búsqueda 3D.

60 14. Un dispositivo de codificación de vídeo que comprende:

una memoria configurada para almacenar datos de vídeo; y

65 uno o más procesadores en comunicación con la memoria, y configurados para:

- 5 generar una tabla de búsqueda tridimensional (3D) para la escalabilidad de gama de colores de los datos de vídeo con particiones más gruesas para los componentes de croma y con particiones más finas para un componente de luma, el uno o más procesadores configurados para particionar cada uno de los componentes de luma, de un primer componente de croma y de un segundo componente de croma de la tabla de búsqueda 3D en un primer número de octantes basándose en un valor de partición base, y
- 10 particionar cada uno del primer número de octantes del componente de luma en un segundo número de octantes basándose en un valor de partición de luma para el componente de luma de la tabla de búsqueda 3D de manera que un número total de octantes para el componente de luma es mayor que un número total de octantes para el primer componente de croma y mayor que un número total de octantes para el segundo componente de croma,
- 15 predecir los bloques de vídeo de los datos de vídeo basándose en al menos una imagen de referencia generada usando la tabla de búsqueda 3D, y
- codificar datos residuales de los bloques de vídeo en un flujo de bits.
15. Un medio de almacenamiento legible por ordenador que almacena instrucciones para decodificar datos de vídeo que, cuando se ejecutan, hacen que uno o más procesadores realicen el procedimiento de cualquiera de las reivindicaciones 1-12.

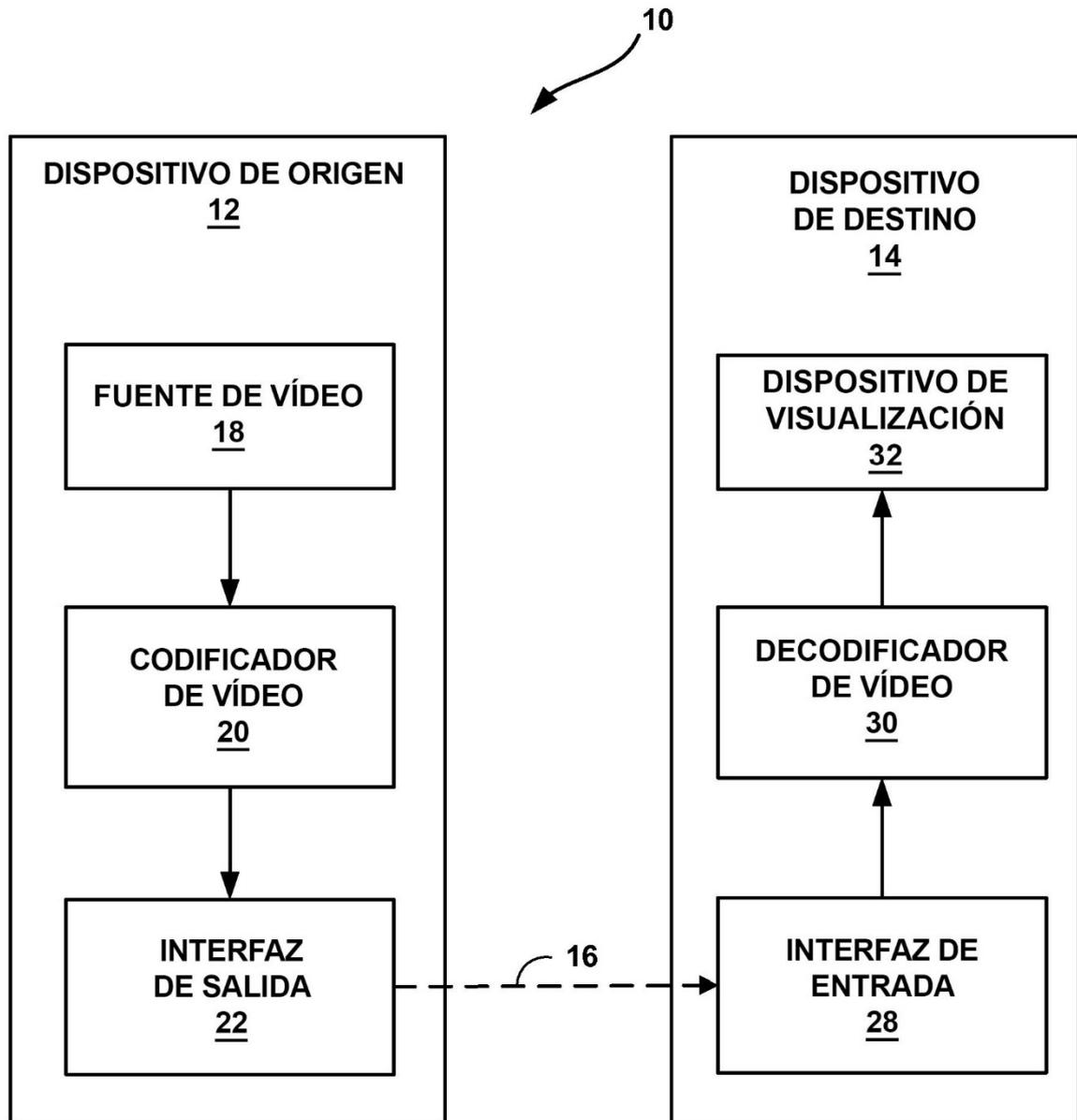


FIG. 1

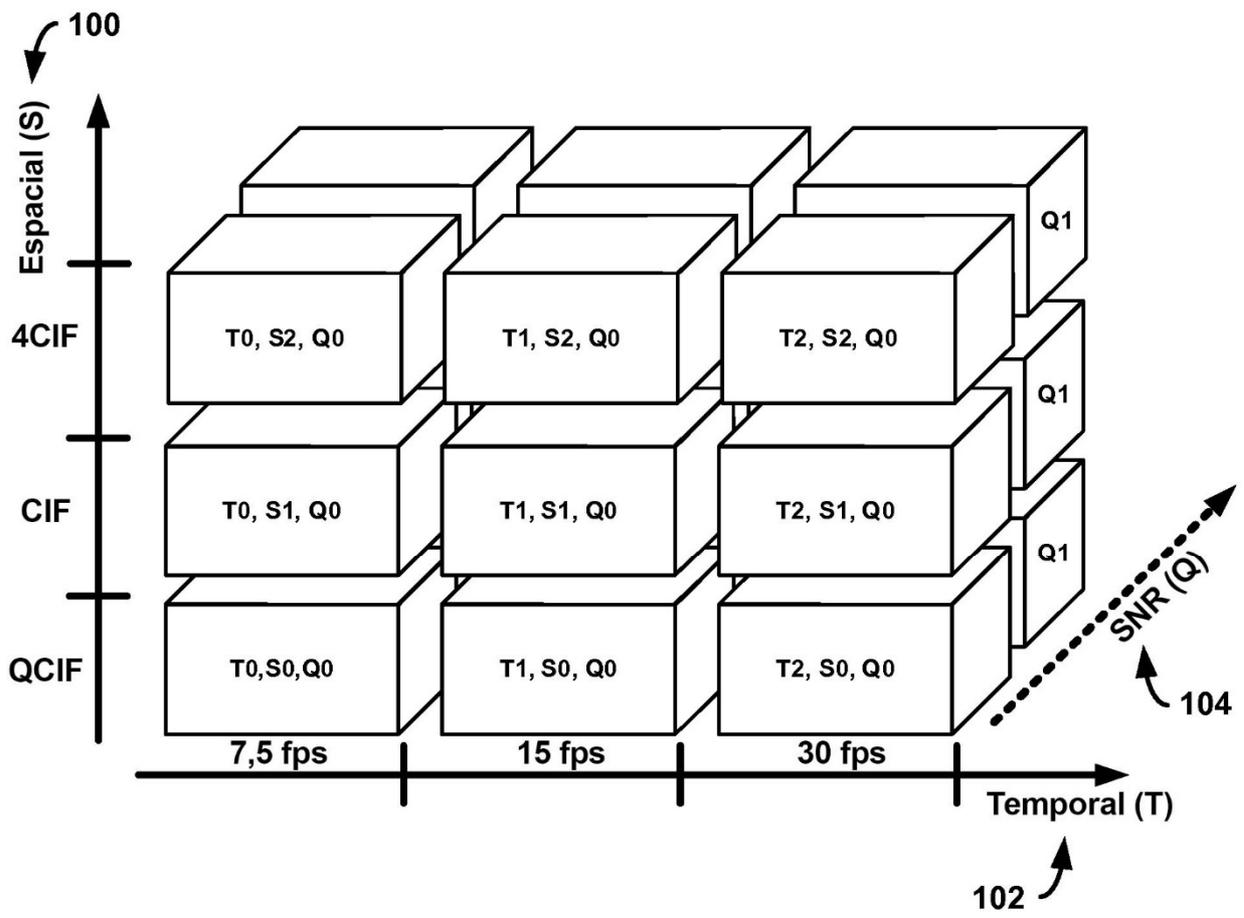


FIG. 2

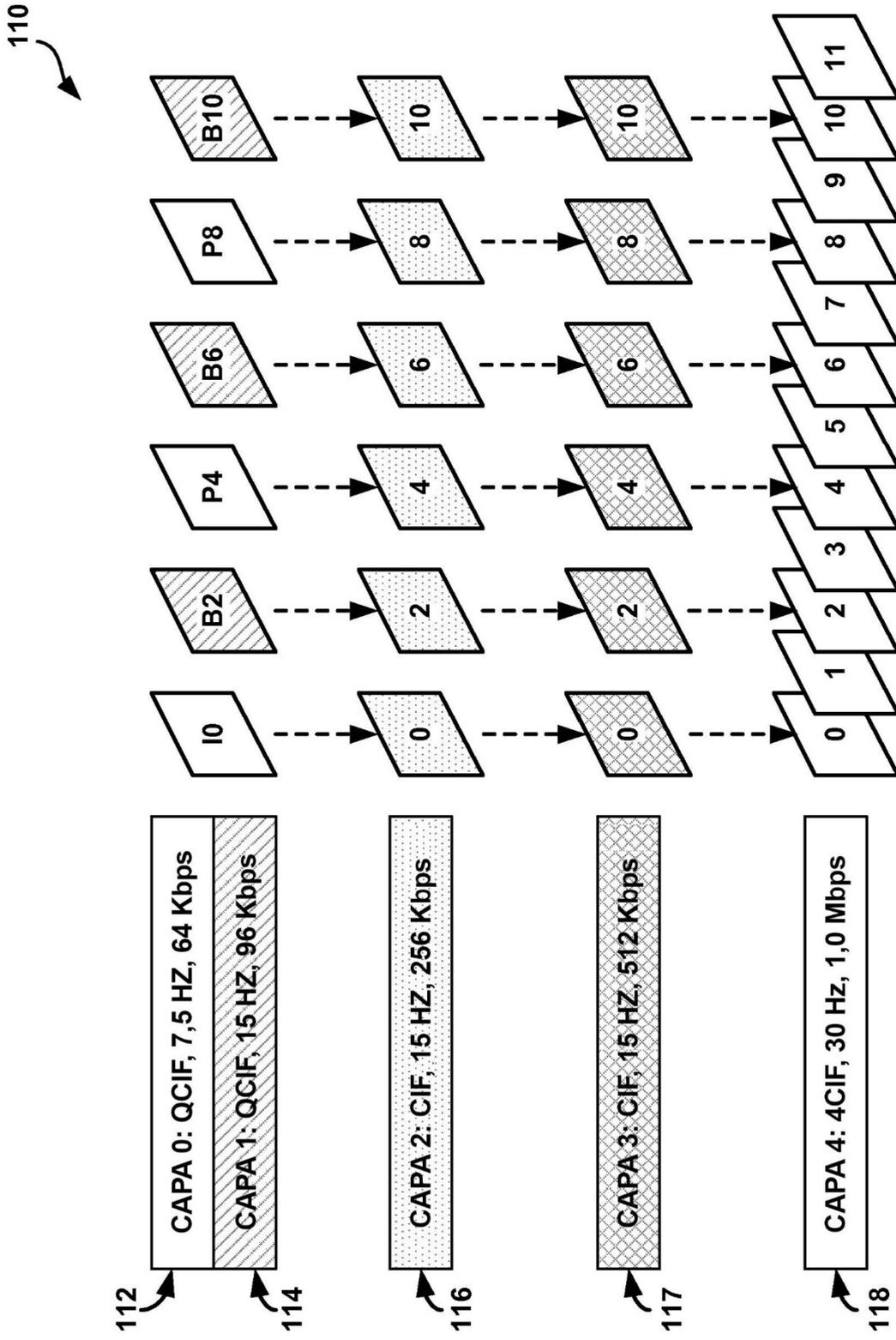


FIG. 3

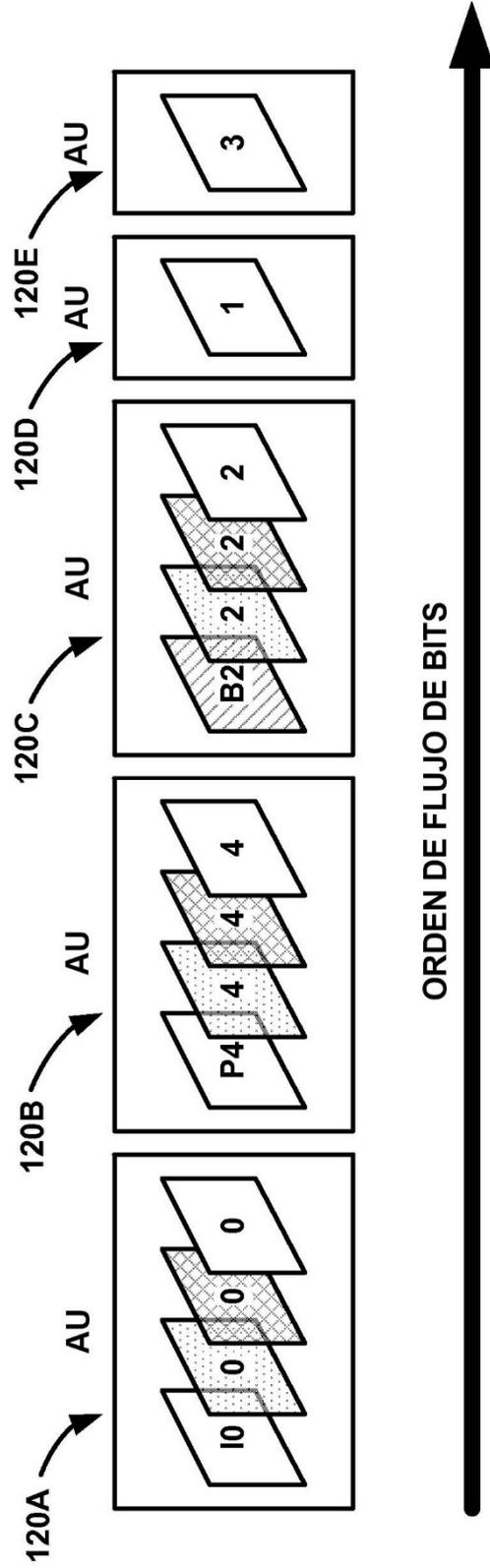


FIG. 4

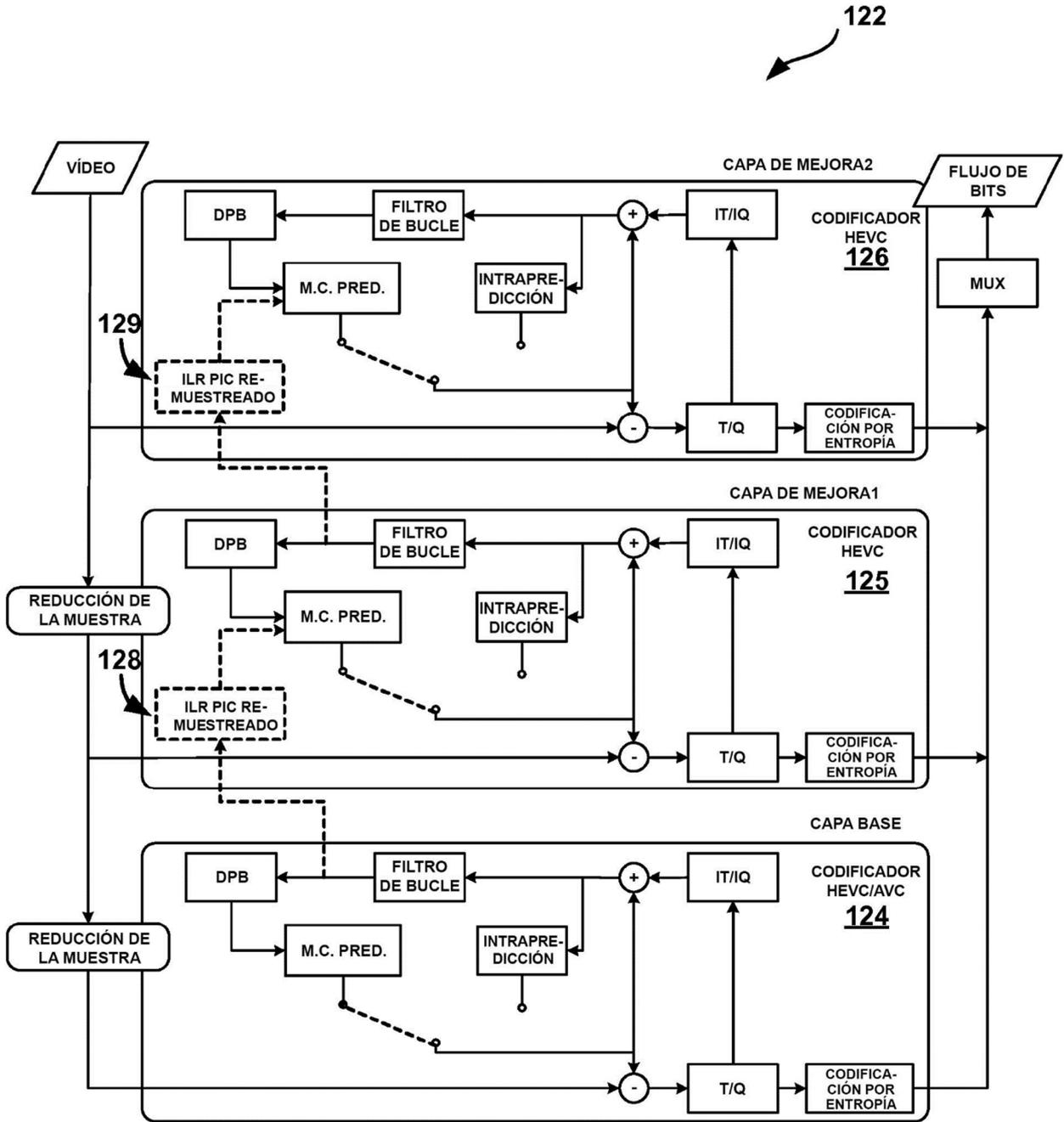


FIG. 5

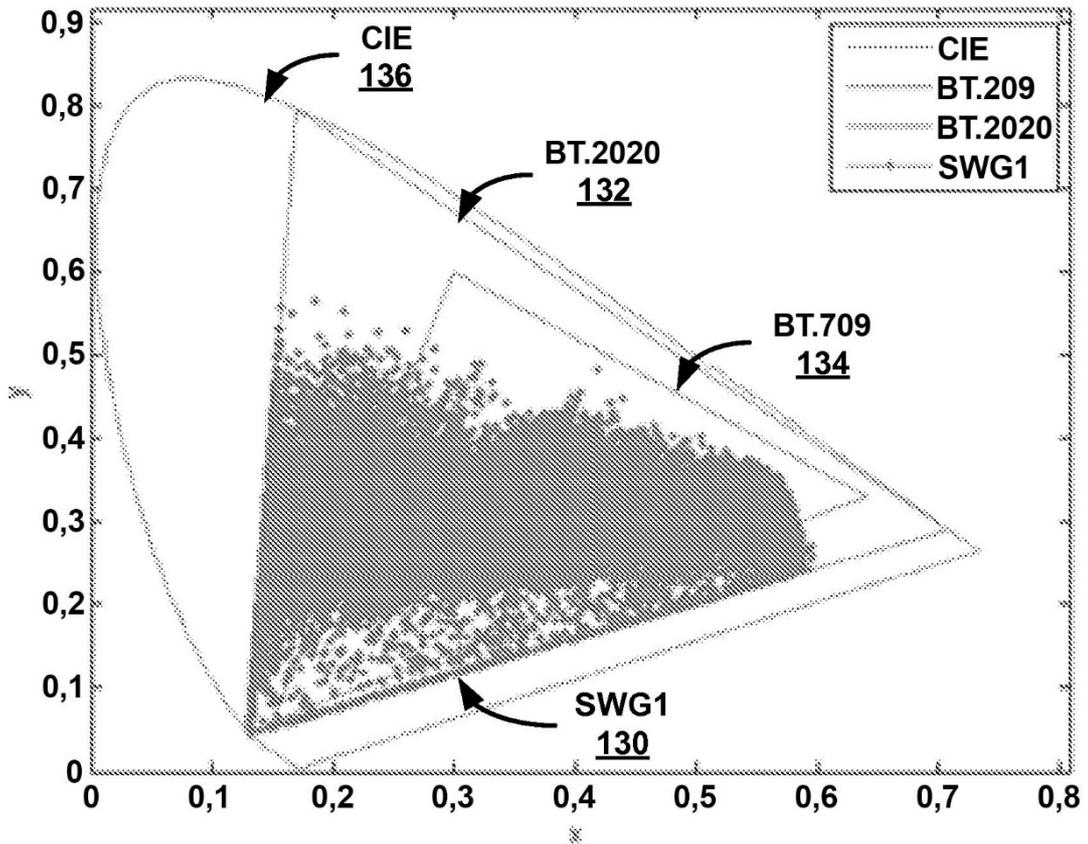


FIG. 6

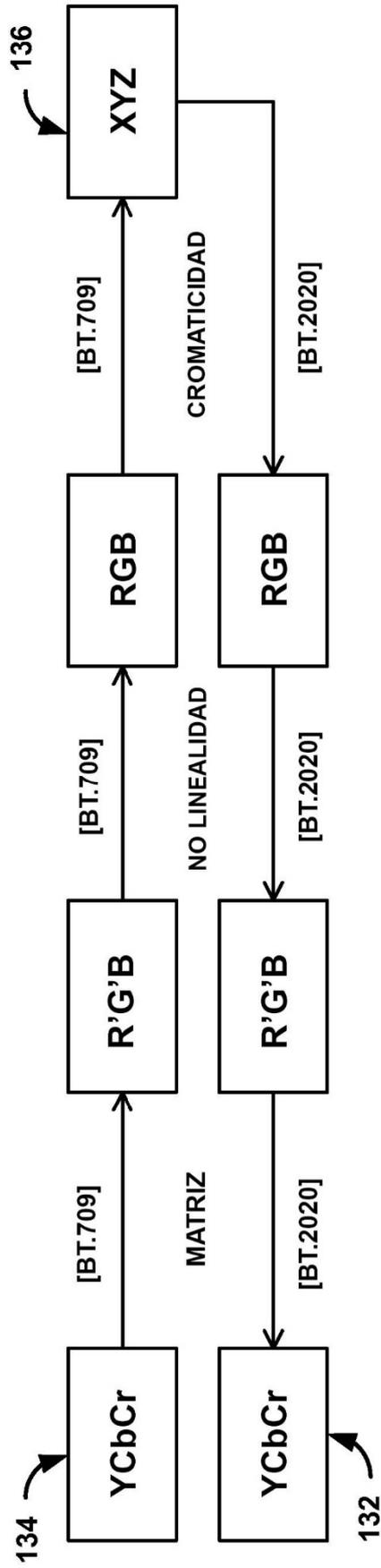


FIG. 7

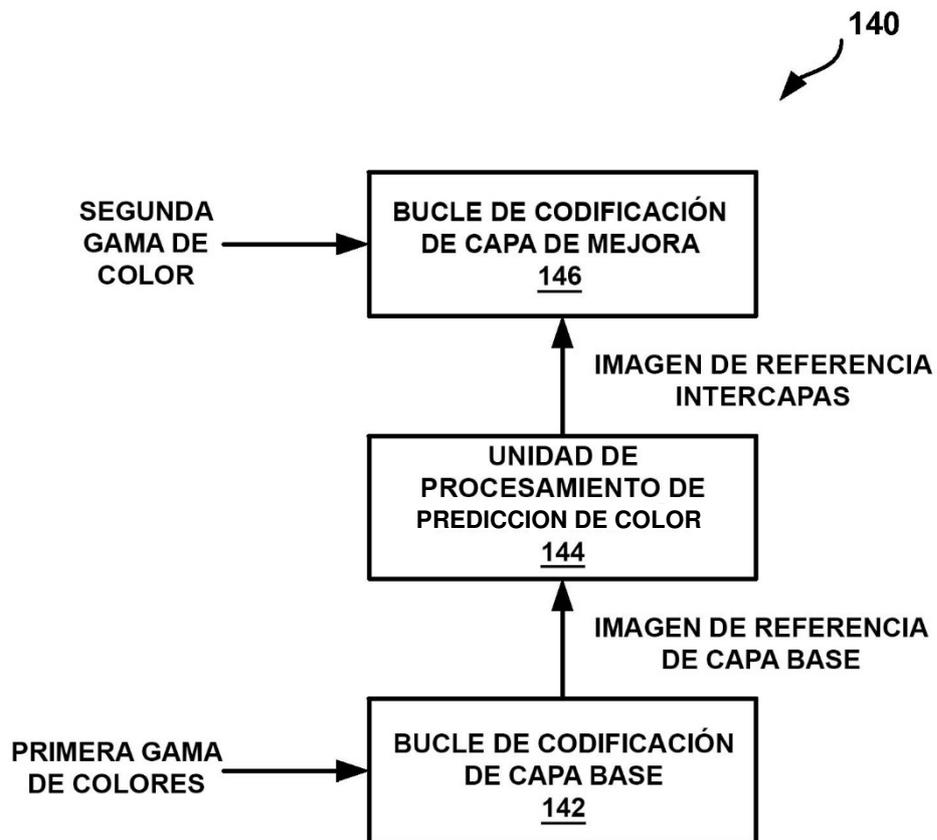


FIG. 8

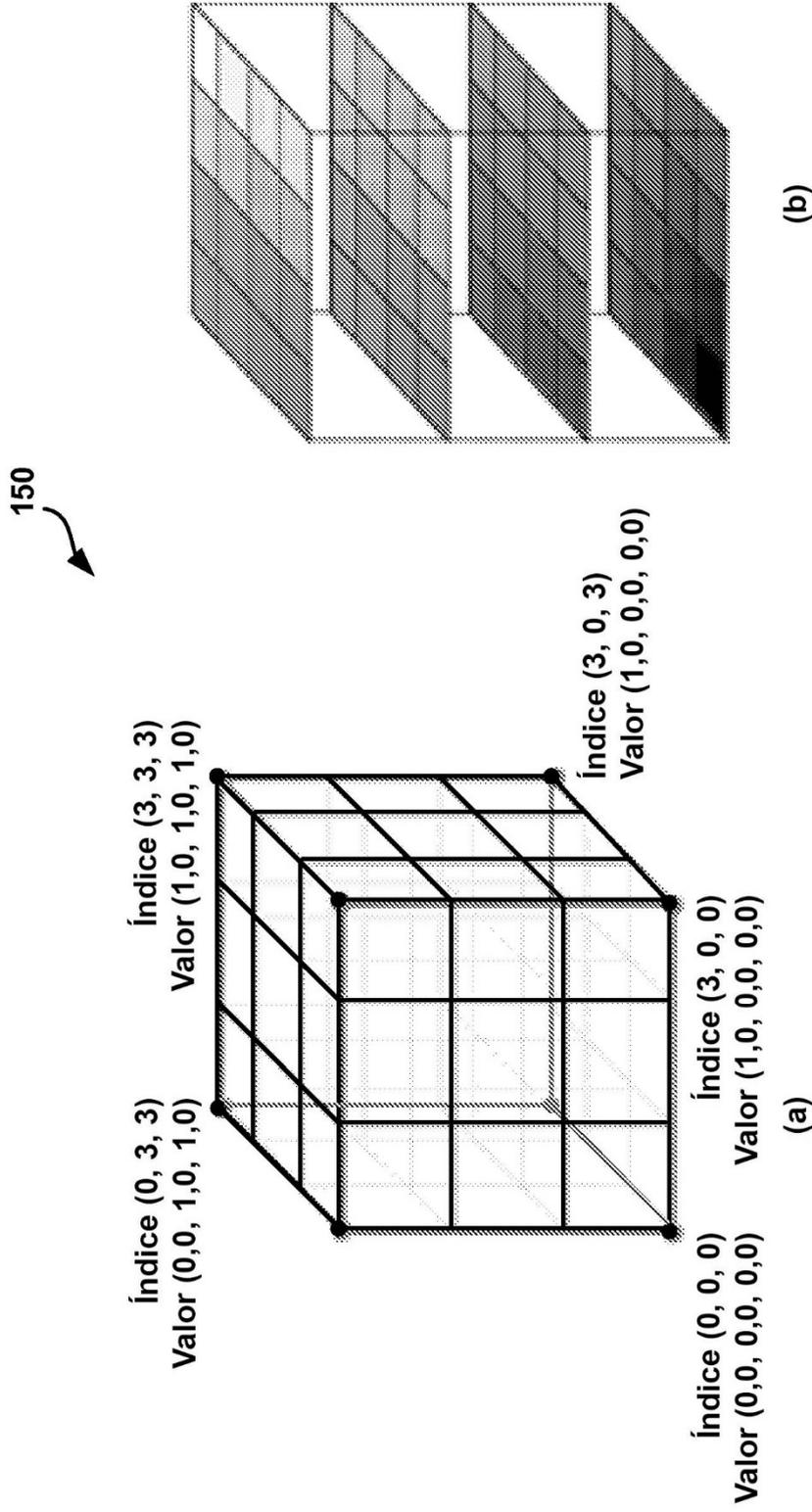


FIG. 9

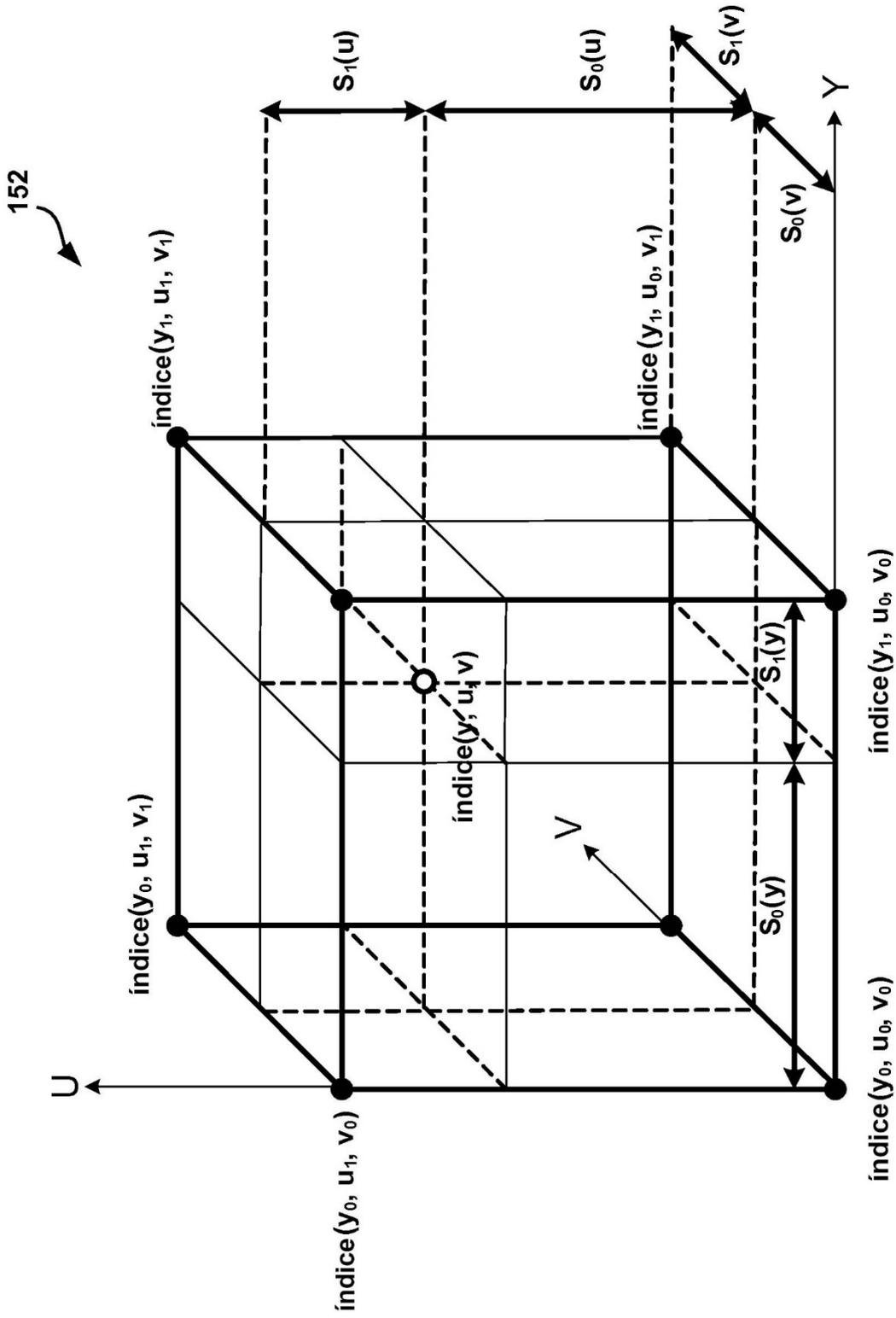


FIG. 10

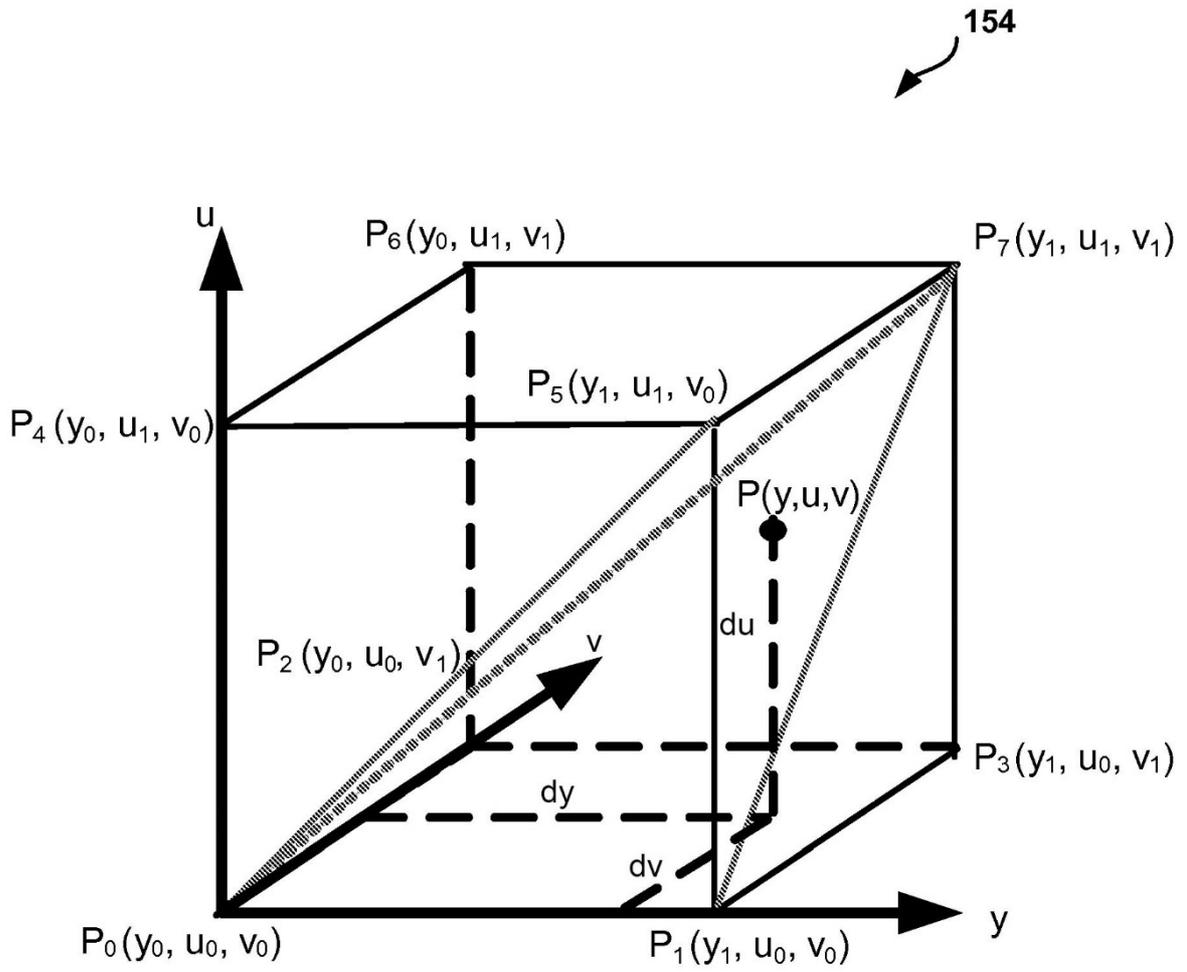


FIG. 11

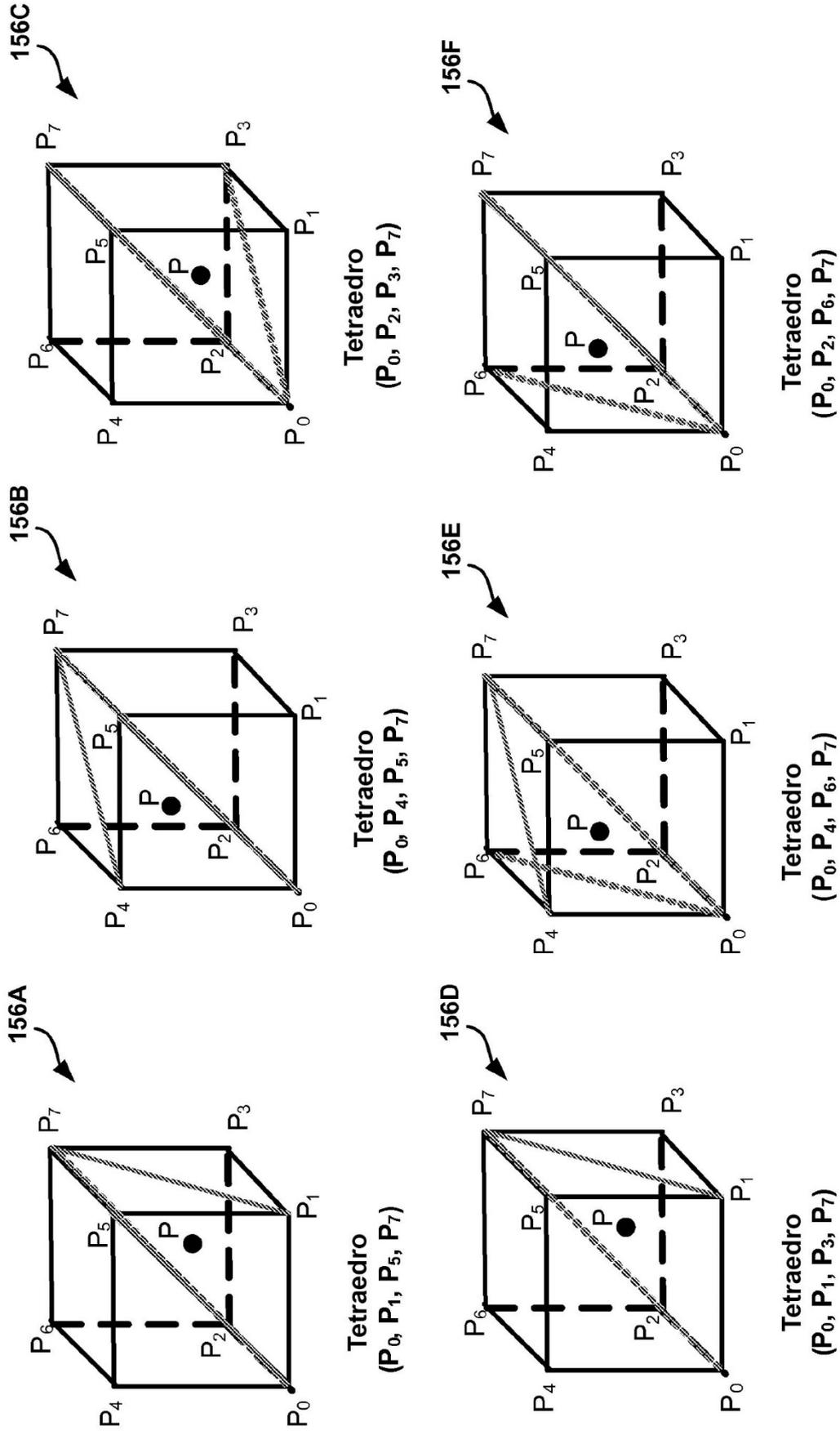


FIG. 12

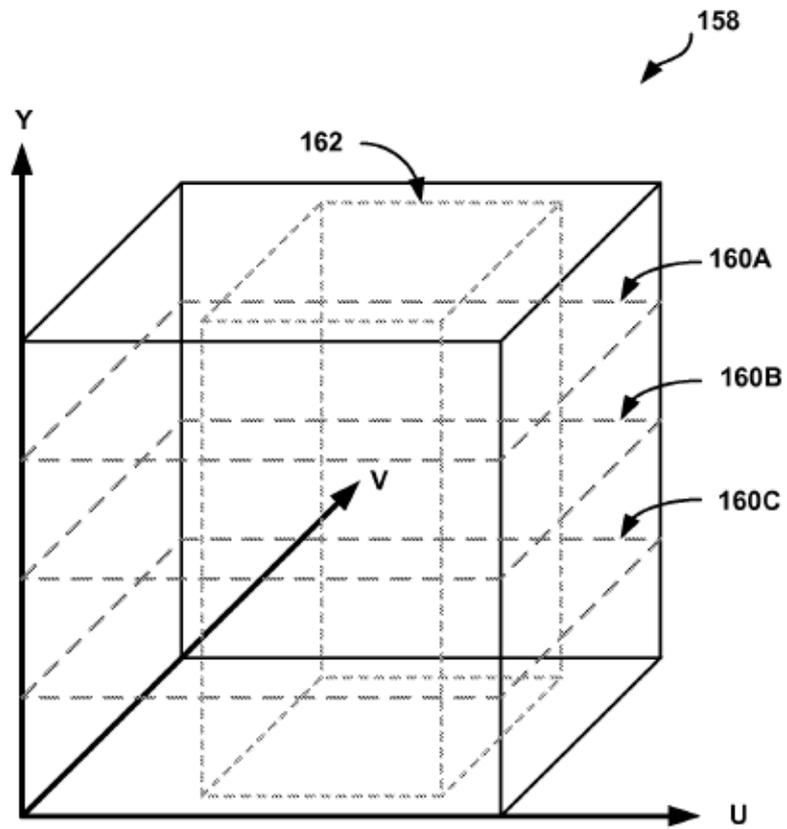


FIG. 13

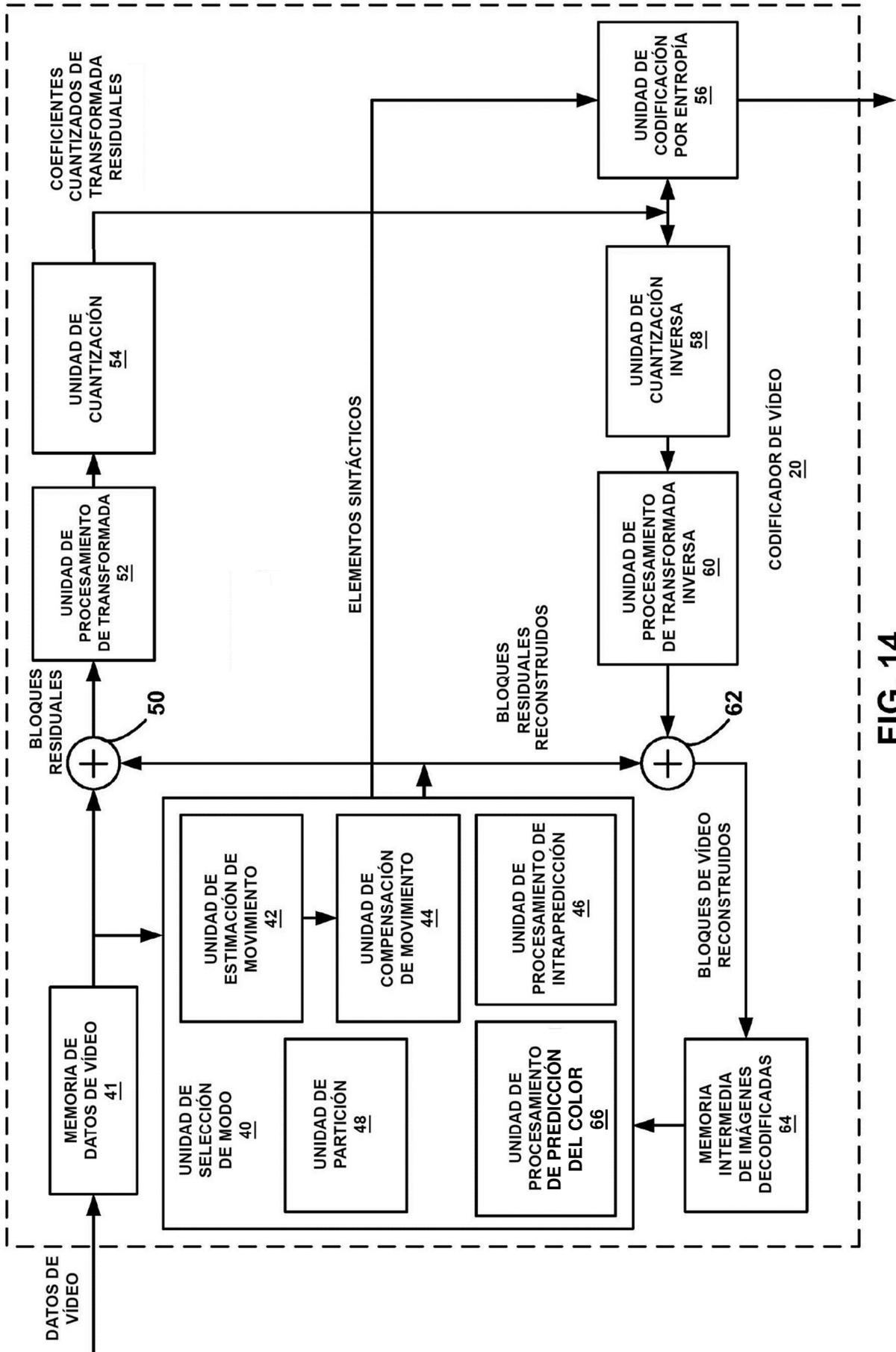


FIG. 14

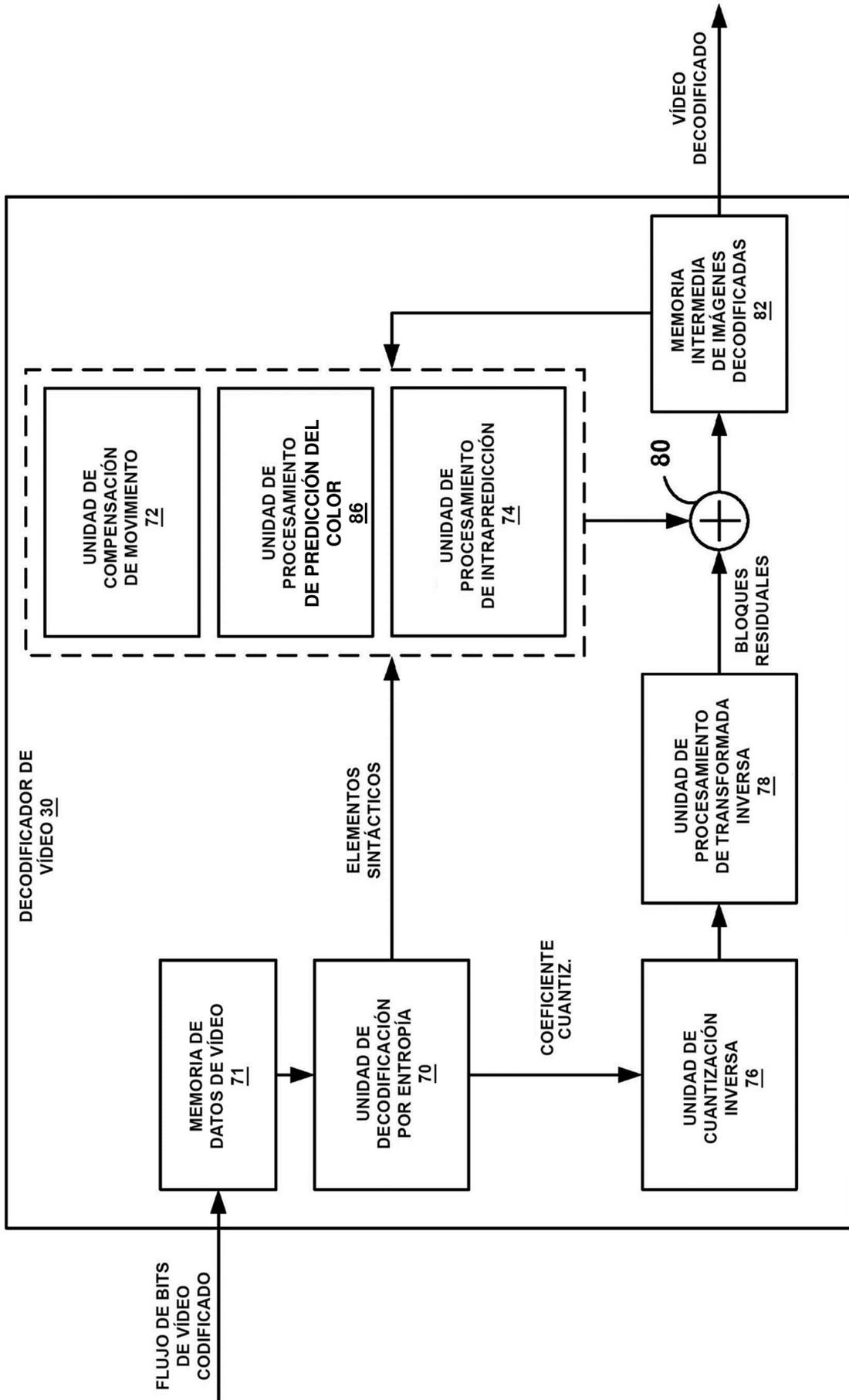


FIG. 15

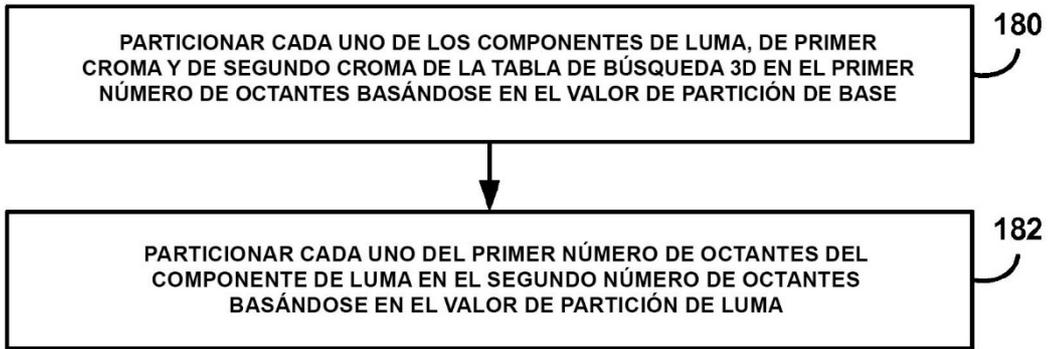


FIG. 16

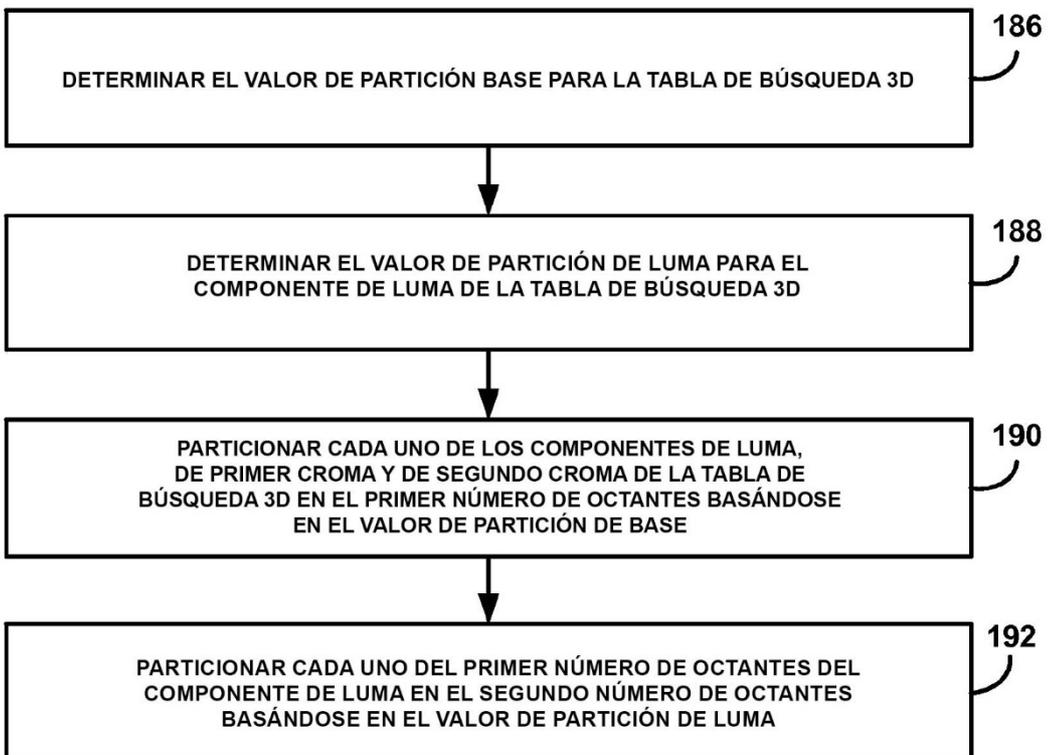


FIG. 17

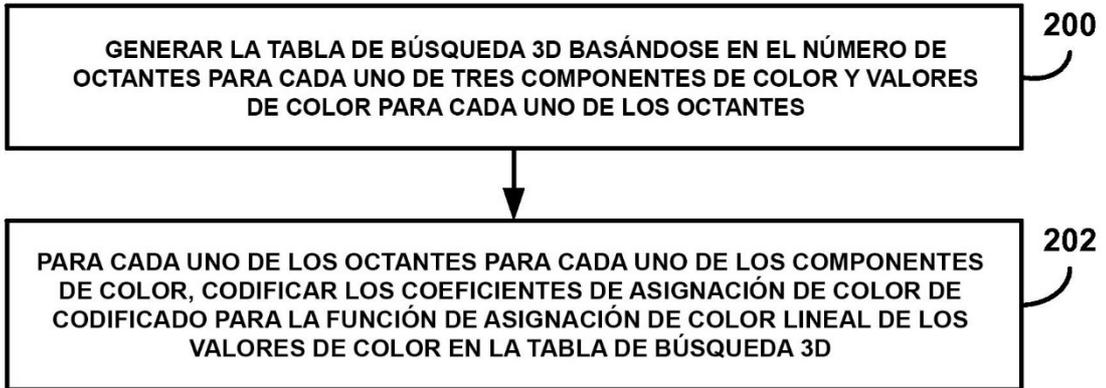


FIG. 18

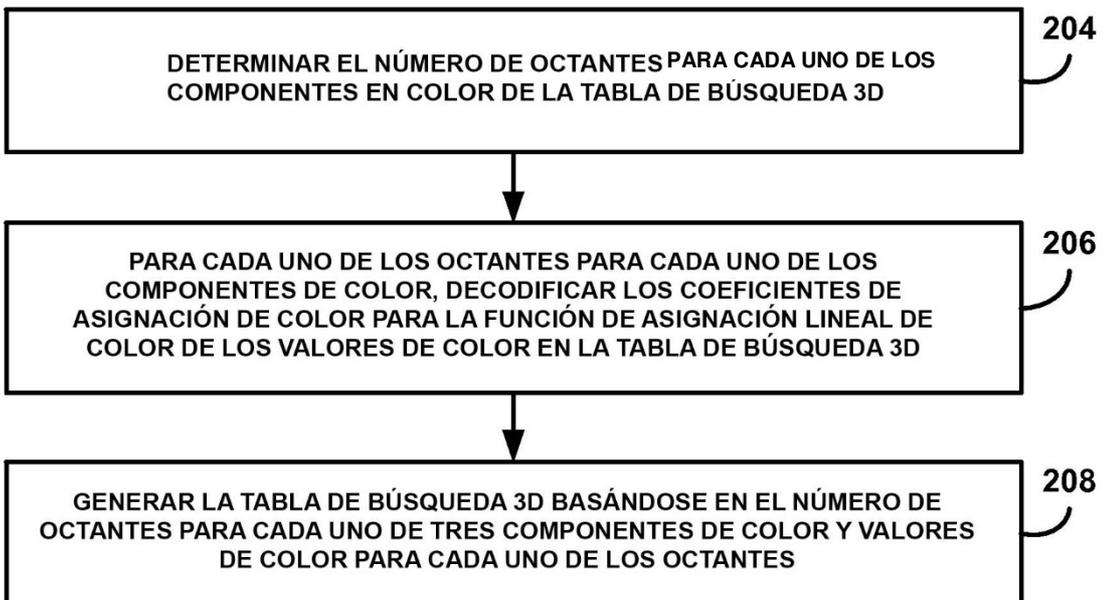


FIG. 19