

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 548 778**

51 Int. Cl.:

H04N 19/597 (2014.01)
H04N 19/70 (2014.01)
H04N 19/172 (2014.01)
H04N 19/46 (2014.01)
H04N 19/61 (2014.01)
H04N 19/174 (2014.01)
H04N 19/44 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **20.07.2012 E 12738703 (3)**

97 Fecha y número de publicación de la concesión europea: **19.08.2015 EP 2735150**

54 Título: **Predicción de cabecera de porciones para mapas de profundidad en códecs de vídeo tridimensionales**

30 Prioridad:

22.07.2011 US 201161510738 P
11.08.2011 US 201161522584 P
26.11.2011 US 201161563772 P
13.04.2012 US 201261624031 P
19.07.2012 US 201213553617

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
20.10.2015

73 Titular/es:

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

72 Inventor/es:

CHEN, YING y
KARCZEWICZ, MARTA

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 548 778 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

Predicción de cabecera de porciones para mapas de profundidad en códecs de vídeo tridimensionales

5 La presente solicitud reivindica el beneficio de las Solicitudes Provisionales de Estados Unidos N° 61/510.738, presentada el 22 de julio de 2011, N° 61/522.584, presentada el 11 de agosto de 2011, N° 61/563.772, presentada el 26 de noviembre de 2011, y N° 61/624.031, presentada el 13 de abril de 2012.

Campo técnico

La presente divulgación se refiere al campo de la codificación de vídeos, por ejemplo, codificación de datos de vídeo tridimensionales.

Antecedentes

10 Las capacidades de vídeo digitales se pueden incorporar en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión directa digitales, dispositivos de comunicación inalámbricos, tales como terminales de radiotelfonos, sistemas de transmisión inalámbricos, asistentes digitales personales (PDA), ordenadores portátiles u ordenadores de escritorio, cámaras digitales, dispositivos de grabación digitales, dispositivos de videojuegos, consolas de videojuegos, y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como MPEG-2, MPEG-4 o H.264/MPEG-4, Parte 10, Codificación de Vídeo Avanzada (AVC), para transmitir y recibir vídeos digitales más eficazmente. Las técnicas de compresión de vídeo realizan una predicción espacial y temporal para reducir o eliminar la redundancia inherente en las secuencias de vídeo.

20 Las técnicas de compresión de vídeo realizan una predicción espacial y/o una predicción temporal para reducir o eliminar la redundancia inherente en las secuencias de vídeo. Para la codificación de vídeos basándose en bloques, un fotograma o franje de vídeo se pueden dividir en macrobloques. Cada macrobloque se puede dividir aún más. Los macrobloques en un fotograma o porción intra-codificado (I) se codifican utilizando predicción espacial con respecto a los macrobloques vecinos. Los macrobloques en un fotograma o porción inter-codificado (P o B) pueden utilizar predicción espacial con respecto a los macrobloques vecinos en el mismo fotograma o una porción o predicción temporal con respecto a otros fotogramas de referencia.

25 Después de que los datos de vídeo se han codificado, los datos de vídeo pueden empaquetarse para su transmisión o almacenamiento. Los datos de vídeo se pueden recopilar en un archivo de vídeo en conformidad con cualquiera de una variedad de normas, tales como formato de archivo multimedia de base de la Organización Internacional de Normalización (ISO) y sus extensiones, tales como AVC.

30 Se han realizado esfuerzos para desarrollar nuevos estándares de codificación de vídeos basándose en H.264/AVC. Uno de tales estándares es el estándar de codificación de vídeos escalable (SVC), que es la extensión escalable de H.264/AVC. Otro estándar es la codificación de vídeos en múltiples vistas (MVC), que se ha convertido en la extensión de múltiples vistas de H.264/AVC. Un proyecto conjunto de la MVC se describe en JVT-AB204, "Proyecto Conjunto 8.0 en Codificación de vídeos de Múltiples Vistas". 28ª sesión JVT, Hannover, Alemania, julio de 2008, disponible en http://wftp3.itu.int/av-arch/JVT-site/2008_07_Hannover/JVT-AB204.zip. Una versión de la norma AVC se describe en JVT-AD007, "Revista del proyecto de los Editores a la Codificación de Vídeo Avanzada UIT-T Rec H.264|ISO/IEC 14496-10 - en preparación para el Consentimiento UIT-T SG 16 AAP (en forma integrada)", "Reunión JVT 30, Ginebra, CH, febrero de 2009", disponible en http://wftp3.itu.int/av-arch/jvt-site/2009_01_Geneva/JVT-AD007.zip. La presente memoria integra SVC y MVC en la especificación de AVC.

Sumario

45 En general, la presente divulgación describe técnicas para soportar procesamiento de vídeos tridimensionales (3D). En particular, las técnicas de la presente divulgación se refieren a la codificación y decodificación de contenidos de vídeos 3D. La presente divulgación propone también técnicas de señalación para unidades de bloque codificadas de datos de vídeo. Por ejemplo, la presente divulgación propone la reutilización de elementos de sintaxis incluidos en una cabecera de porciones de componentes de vista de textura para componentes de vista de profundidad correspondientes. Adicionalmente, la presente divulgación propone la reutilización de elementos de sintaxis en la información de cabecera de porciones de componentes de vista de profundidad para componentes de vista de textura.

50 En un codec 3D, un componente de vista de cada vista de datos de vídeo en un instante de tiempo específica puede incluir un componente de vista de textura y un componente de vista de profundidad. El componente de vista de textura puede incluir componentes de luminancia (Y) y componentes de crominancia (Cb y Cr). Los componentes de luminancia (brillo) y crominancia (color) se denominan colectivamente en la presente memoria como componentes "textura". El componente de vista de profundidad puede ser un mapa de profundidad de una imagen. En la representación de imágenes en 3D, los mapas de profundidad incluyen componentes de profundidad que son representativos de valores de profundidad, por ejemplo, para los componentes de textura correspondientes. Los componentes de vista de profundidad se pueden utilizar para generar vistas virtuales desde una perspectiva de vista

proporcionada.

Los elementos de sintaxis para los componentes de profundidad y los componentes de textura pueden señalarse con una unidad de bloque codificado. Las unidades de bloques codificados, también conocido simplemente como "bloques codificados" en la presente divulgación, pueden corresponder a macrobloques en UIT-T H.264/AVC (Codificación de Vídeo Avanzada) o unidades de codificación de Codificación de vídeos de Alta Eficiencia (HEVC).

En un aspecto, un procedimiento de decodificación incluye la recepción de una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. El procedimiento incluye además recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. El procedimiento comprende además decodificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende todos los elementos de sintaxis representativos de las características de la primera porción y la determinación de los elementos de sintaxis comunes para una segunda porción de la cabecera de porciones de la primera porción. El procedimiento puede incluir además la decodificación de la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

En otro aspecto, un dispositivo para la decodificación de datos incluye un decodificador de vídeo configurado para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura, recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las de características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso, decodificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende todos los elementos de sintaxis representativos de las características de la primera porción, determinar elementos de sintaxis comunes para una segunda porción de la cabecera de porciones de la primera porción, y decodificar la segunda porción después de la decodificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

En otro aspecto, un producto de programa informático comprende un medio de almacenamiento legible por ordenador que tiene almacenado en su interior instrucciones que, al ejecutarse, hacen que un procesador de un dispositivo de decodificación de vídeos reciba una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. Las instrucciones hacen adicionalmente que el procesador del dispositivo de decodificación de vídeos reciba una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en las que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en las que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. Las instrucciones hacen adicionalmente que el procesador del dispositivo de decodificación de vídeos decodifique una primera porción, en las que la primera porción comprende una de la porción de textura y de la porción de profundidad, en las que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción y determine los elementos de sintaxis comunes para una segunda porción de la cabecera de porciones de la primera porción. Las instrucciones hacen adicionalmente que el procesador del dispositivo de decodificación de vídeos decodifique la segunda porción después de la decodificación de la primera porción, al menos parcialmente basándose en los

elementos de sintaxis comunes determinados, en las que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en las que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

5 En otro aspecto, se proporciona un dispositivo que comprende medios para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. El dispositivo comprende además medios para recibir una porción de profundidad para un componente de
10 vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. El dispositivo
15 comprende además medios para decodificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción. El dispositivo comprende además medios para decodificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción
20 comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en el que el segundo porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

25 En un aspecto, un procedimiento de codificación incluye la recepción de una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. El procedimiento incluye además recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de
30 vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. El procedimiento comprende además codificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la
35 porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción y determinar los elementos de sintaxis comunes para un segundo porción de la cabecera de porciones de la primera porción. El procedimiento puede incluir, además, codificar la segunda porción después de codificar la primera porción, al menos parcialmente basándose en los elementos de sintaxis común determinados, en el que la segunda porción comprende una de la porción de textura y
40 porción de profundidad diferentes de la primera porción, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

45 En otro aspecto, un dispositivo para la codificación de datos incluye un codificador de vídeo configurado para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura, recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de
50 la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. El codificador de vídeo se configura además para codificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que
55 comprende elementos de sintaxis representativos de las características de la primera porción, determinar los elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción, y codificar la segunda porción después de codificar la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción,
60 excluyendo los valores para los elementos de sintaxis comunes a la primera porción.

En otro aspecto, un producto de programa informático comprende un medio de almacenamiento legible por ordenador que tiene almacenado en su interior instrucciones que, al ejecutarse, hacen que un procesador de un

dispositivo de codificación de vídeos reciba una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. Las instrucciones hacen
 5 adicionalmente que el procesador del dispositivo de codificación de vídeos reciba una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en las que la porción de profundidad comprende el uno o más bloques codificados de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en las que el
 10 componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. Las instrucciones hacen adicionalmente que el procesador del dispositivo de codificación de vídeos codifique una primera porción, en las que la primera porción comprende una de la porción de textura y de la porción de profundidad, en las que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción y determine los elementos de sintaxis comunes para una segunda porción de la cabecera de porciones de la primera porción. Las instrucciones hacen
 15 adicionalmente que el procesador del dispositivo de codificación de vídeos codifique la segunda porción después de la decodificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en las que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en las que la segunda porción tiene una cabecera de porciones que comprende
 20 elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

En otro aspecto, se proporciona un dispositivo que comprende medios para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de
 25 porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. El dispositivo comprende además medios para recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en la que la porción de profundidad comprende el uno o más bloques codificados de la información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción de profundidad, y en el que el componente de vista de
 30 profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso. El dispositivo comprende además medios para decodificar una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción. El dispositivo
 35 comprende además medios para determinar los elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción. El dispositivo comprende además medios para codificar la segunda porción después de terminar con la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y porción de profundidad diferentes de la primera porción, en el que el segundo porción tiene una cabecera de porciones que
 40 comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

Las técnicas descritas en la presente divulgación se pueden implementar en hardware, software, firmware, o cualquier combinación de los mismos. Si se implementan en software, el software se puede ejecutar en un procesador, que se puede referir a uno o más procesadores, tal como un microprocesador, aplicación de circuito
 45 integrado específico (ASIC), campo de matriz de compuertas programable (FPGA), o procesador de señal digital s(DSP), u otro circuitos lógicos integrados o discretos equivalentes. El software que comprende instrucciones para ejecutar las técnicas se puede almacenar inicialmente en un medio legible por ordenador y cargarse y ejecutarse por un procesador.

Por consiguiente, la presente divulgación contempla también medios legibles por ordenador que comprenden instrucciones para hacer que un procesador realice cualquiera de una variedad de técnicas como se describe en la
 50 presente divulgación. En algunos casos, el medio legible por ordenador puede formar parte de un programa informático, que podrá venderse a los fabricantes y/o utilizarse en un dispositivo. El producto de programa informático puede incluir el medio legible por ordenador, y en algunos casos, puede incluir también materiales de empaquetado.

La presente divulgación se puede aplicar también a señales electromagnéticas que transmiten información. Por ejemplo, una señal electromagnética puede comprender información relacionada con todo el soporte de píxel utilizado para interpolar un valor de un píxel sub-entero de una muestra de referencia. En algunos ejemplos, una
 55 señal se puede generar a partir de o transmitirse por un dispositivo que implemente las técnicas descritas en la presente memoria. En otros ejemplos, la presente divulgación se puede aplicar a las señales que se pueden recibir en un dispositivo que implemente las técnicas descritas en la presente memoria.

Los detalles de uno o más aspectos de la divulgación se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetos, y ventajas de las técnicas descritas en la presente divulgación serán

evidentes a partir de la descripción y los dibujos, y de las reivindicaciones.

Breve descripción de los dibujos

La Figura 1 es un diagrama de bloques que ilustra un ejemplo de un sistema de codificación y decodificación de vídeos, de acuerdo con las técnicas de la presente divulgación.

5 La Figura 2 es un diagrama de bloques que ilustra un ejemplo del codificador de vídeo de la Figura 1 con más detalle, de acuerdo con las técnicas de la presente divulgación.

La Figura 3 es un diagrama de un ejemplo de una estructura de predicción de MVC para codificación de vídeos en múltiples vistas, de acuerdo con las técnicas de la presente divulgación.

10 La Figura 4 es un diagrama de flujo que ilustra una operación ejemplar de un codificador de vídeo, de acuerdo con las técnicas de la presente divulgación.

La Figura 5 es un diagrama de bloques que ilustra un ejemplo del decodificador de vídeo de la Figura 1 con más detalle, de acuerdo con las técnicas de la presente divulgación.

La Figura 6 es un diagrama de flujo que ilustra una operación ejemplar de un decodificador de vídeo, de acuerdo con las técnicas de la presente divulgación.

Descripción detallada

La presente divulgación describe técnicas de señalación que un codificador puede aplicar y que un decodificador puede utilizar durante al menos la etapa inter-predicción de al menos un procedimiento de codificación o decodificación de vídeos. Las técnicas descritas están relacionadas con la codificación de contenido de vídeo tridimensional ("3D"). El contenido de vídeo en 3D se puede representar, por ejemplo, como bloques codificados de profundidad más vídeo en múltiples vistas ("MVD"). Es decir, estas técnicas se pueden aplicar para codificar o decodificar una cadena de bits que representa una cadena de bits (MVC) de codificación de vídeos en múltiples vistas, donde cualquiera o todas las vistas de la cadena de bits MVC pueden incluir además información de profundidad.

25 Más específicamente, algunas técnicas de acuerdo con la presente divulgación implican la recepción de al menos una imagen bidimensional que tienen componentes de vista de textura y componentes de vista de profundidad. Algunos componentes de vista de textura y de los componentes de vista de profundidad se pueden codificar en un solo bloque codificado o como bloques separados. Una imagen se puede dividir en porciones de imágenes. Los elementos de sintaxis para la codificación de los componentes de vista de textura se pueden señalar en una cabecera de porciones. Algunos elementos de sintaxis para los componentes de vista de profundidad se pueden predecir a partir de los elementos de sintaxis para los componentes de vista de textura correspondientes a los componentes de vista de profundidad. Las técnicas de la presente divulgación se refieren a la codificación, decodificación, y la señalación de los datos utilizados para representar datos de vídeo tridimensionales a partir de datos de vídeo bidimensionales, basándose en datos de mapa de profundidad estimados para los datos de vídeo bidimensionales. En algunos ejemplos, los componentes de vista de textura se codifican utilizando diferentes técnicas a las utilizadas para la codificación de la información de profundidad. En la presente divulgación, el término "codificación" se puede referir a uno o ambos de la codificación y decodificación.

La conversión de vídeo basándose en la estimación de la profundidad y la síntesis de vista virtual se utiliza para crear imágenes en 3D, tal como para aplicaciones de vídeo en 3D. En particular, las vistas virtuales de una escena se pueden utilizar para crear una vista 3D de la escena. La generación de una vista virtual de una escena basándose en una vista existente de la escena se logra convencionalmente mediante la estimación de los valores de profundidad objeto antes de sintetizar la vista virtual. La estimación de profundidad es un procedimiento de estimación de distancias absolutas o relativas entre los objetos y un plano de la cámara a partir de pares estéreo o contenido monoscópico. Tal como se utiliza en la presente memoria, la información de profundidad incluye información útil en la formación de vídeo tridimensional, tal como un mapa de profundidad (por ejemplo, valores de profundidad sobre una base por píxel) o un mapa de paralaje (por ejemplo, la disparidad horizontal sobre una base por píxel).

La información de la profundidad estimada, representada por lo general por un mapa de profundidad de la imagen al nivel de grises, se puede utilizar para generar un ángulo arbitrario para las vistas virtuales utilizando técnicas de representación basándose en imágenes de profundidad (DIBR). En comparación con los sistemas de televisión tridimensionales (3DTV) tradicionales donde secuencias de múltiples vistas se enfrentan a los retos de la compresión inter- vista eficaz, un sistema basándose en mapas de profundidad puede reducir el uso de ancho de banda mediante la transmisión de solo una o unas pocas vistas junto con el mapa o mapas de profundidad mapa, que se puede codificar de manera eficaz. El mapa o mapas de profundidad utilizados en la conversión basándose en mapas de profundidad se puede controlar (por ejemplo, a través de escalado) por los usuarios finales antes de que el mapa o mapas de profundidad se utilicen en la síntesis de vista. Las vistas virtuales personalizadas se pueden generar con diferente cantidad de profundidad percibida. Además, una estimación de la profundidad se puede realizar utilizando vídeo monoscópico en el que solo hay disponible un contenido de vista en 2D.

Las técnicas descritas en la presente memoria se pueden aplicar para predecir elementos de sintaxis para un componente de vista de profundidad a partir de los elementos de sintaxis almacenados en una cabecera de porciones

para los componentes de vista de textura de ubicación conjunta de la misma vista. Por ejemplo, los valores de los elementos de sintaxis comunes en la porción de profundidad y en la porción de textura se pueden incluir en la cabecera de porciones para los componentes de vista de textura, pero no en la porción para los componentes de vista de profundidad asociados. Es decir, un codificador o decodificador de vídeo pueden codificar los elementos de sintaxis comunes en la porción de profundidad y en porción de textura en la cabecera de porciones para los componentes de vista de textura que no están presentes en la cabecera de porciones para los componentes de vista de profundidad. Por ejemplo, un primer valor se puede proporcionar para un primero elemento de sintaxis en la cabecera de porciones para los componentes de vista de textura. La cabecera de porciones de los componentes de vista la profundidad comparten también el primer elemento de sintaxis, es decir, el primer elemento de sintaxis es común tanto para la cabecera de porciones de textura como para la cabecera de porciones de profundidad. El primer elemento de sintaxis para los componentes de vista de profundidad tiene un segundo valor. Sin embargo, la cabecera de porciones para el componente de vista de profundidad no incluye el primer elemento de sintaxis. De acuerdo con las técnicas descritas en la presente memoria, el segundo valor del primer elemento de sintaxis se puede predecir a partir del primer valor.

En algunos ejemplos, solo una identificación del conjunto de parámetros de imagen (PPS) y un parámetro de cuantificación delta (QP) de una porción se señalan para la cabecera de porciones del componente de vista de profundidad. En otros ejemplos, más información de la lista de imágenes de referencia de la construcción se señala, además de una identificación PPS y delta QP. Otros elementos de sintaxis se derivan o determinan a partir de la cabecera de porciones del componente de vista de textura. En algunos ejemplos, los valores de los elementos de sintaxis comunes están configurados para ser iguales a los elementos de sintaxis correspondientes. Es decir, los demás elementos de sintaxis para la cabecera de porciones del componente de vista de profundidad se establecen iguales a los valores correspondientes en la cabecera de porciones para el componente de vista de textura correspondiente.

En otro ejemplo, la posición de partida del bloque codificado (macrobloque o unidad de codificación) se señala adicionalmente. Es decir, la cabecera de porciones para una porción de información de profundidad indica la ubicación del primer bloque (por ejemplo, primer macrobloque o CU) de la porción, sin señalar otros datos de sintaxis para la cabecera de porciones (que se pueden determinar como iguales a los datos de sintaxis correspondientes de la porción que incluye la información de textura correspondiente). Cuando no se indica la posición de partida de la porción, se infiere que es 0 en algunos ejemplos. Un valor `frame_num` y POC del componente de vista la profundidad se puede señalar adicionalmente. Un indicador se utiliza para indicar si uno o más parámetros de filtro de bucle utilizados para el componente de vista de profundidad son los mismos que uno o más parámetros de filtro de bucle señalados para los componentes de vista de textura.

La inter-codificación basándose en bloques es una técnica de codificación que se basa en la predicción temporal para reducir o eliminar la redundancia temporal entre bloques de vídeo de unidades codificadas sucesivas de una secuencia de vídeo. Las unidades codificadas pueden comprender fotogramas de vídeo, porciones de fotogramas de vídeo, grupos de imágenes u otra unidad definida de bloques de vídeo codificados. Para la inter-codificación, un codificador de vídeo realiza la estimación de movimiento y la compensación de movimiento para estimar el movimiento entre los bloques de vídeo de dos o más unidades codificadas adyacentes. Al utilizar técnicas de estimación de movimiento, el codificador de vídeo genera vectores de movimiento, que indican el desplazamiento de bloques de vídeo relativos a los bloques de vídeo de predicción correspondientes en uno o más fotogramas de referencia u otras unidades codificadas. Al utilizar técnicas para la compensación de movimiento, el codificador de vídeo utiliza los vectores de movimiento para generar bloques de vídeo de predicción a partir del uno o más fotogramas de referencia u otras unidades codificadas. Después de la compensación de movimiento, el codificador de vídeo calcula los bloques de vídeo residuales restando los bloques de vídeo de predicción de los bloques de vídeo originales que se codifican.

Los componentes de vistas de referencia (RVCS) pueden incluir múltiples porciones de textura o profundidad. En algunos ejemplos, donde los componentes vista de referencia comprenden múltiples porciones, se puede utilizar una porción de ubicación conjunta cuando se determinan los elementos de sintaxis de una porción actual. Como alternativa, una primera porción en el RVC se puede utilizar para determinar los elementos de sintaxis de la porción actual. En otros ejemplos, otra porción en el RVC se puede utilizar para determinar los elementos de sintaxis comunes de la porción actual.

La Figura 1 es un diagrama de bloques que ilustra un ejemplo de un sistema 10 de codificación y decodificación de vídeos, de acuerdo con las técnicas de la presente divulgación. Como se muestra en el ejemplo de la Figura 1, el sistema 10 incluye un dispositivo 12 de origen que transmite vídeo codificado a un dispositivo 14 de destino a través de un enlace 15. El enlace 15 puede comprender cualquier tipo de medio o dispositivo capaz de mover los datos de vídeo codificados del dispositivo 12 de origen al dispositivo 14 de destino. En un ejemplo, el enlace 15 comprende un medio de comunicación para permitir que el dispositivo 12 de origen transmita datos de vídeo codificados directamente al dispositivo 16 de destino en tiempo real. Los datos de vídeo codificados se pueden modular de acuerdo con un estándar de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo 16 de destino

El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o por cable, tales como un espectro por radiofrecuencia (RF) o una o más líneas de transmisión físicas. El medio de comunicación puede formar parte de una red basándose en paquetes, tal como una red de área local, una red de área amplia, o una red global como Internet. El medio de comunicación puede incluir enrutadores, conmutadores, estaciones base, o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo 12 de origen al dispositivo 16 de destino.

El dispositivo 12 de origen y el dispositivo 14 de destino pueden comprender cualquiera de una amplia gama de dispositivos. En algunos ejemplos, uno o ambos del dispositivo 12 de origen y el dispositivo 14 de destino pueden comprender dispositivos de comunicación inalámbricos, tales como terminales inalámbricos, los denominados radiotelefonos móviles o por vía satélite, o cualquier dispositivo inalámbrico que pueda comunicar información de vídeo a través del enlace 15, en cuyo caso el enlace 15 es inalámbrico. Las técnicas de la presente divulgación, sin embargo, que se refieren a bloques de codificación de datos de vídeo que incluyen tanto información de textura como de profundidad, no se limitan necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas también pueden ser útiles en una amplia gama de otras configuraciones y dispositivos, incluyendo dispositivos que se comunican a través de cables físicos, fibras ópticas u otros medios físicos o inalámbricos. Además, las técnicas de codificación o decodificación se pueden aplicar también en un dispositivo independientemente de que no se comunique necesariamente con cualquier otro dispositivo. Por ejemplo, el decodificador 28 de vídeo puede residir en un reproductor multimedia digital u otro dispositivo y recibir datos de vídeo codificados a través de medios de streaming, descarga o almacenamiento. Por lo tanto, la representación del dispositivo 12 de origen y del dispositivo 14 de destino en comunicación uno con el otro se proporciona para fines de ilustración de un ejemplo de implementación, y no se debe considerarse limitante en cuanto a las técnicas descritas en la presente divulgación, que pueden ser aplicable a la codificación de vídeos, por lo general, en una variedad de entornos, aplicaciones o implementaciones.

En el ejemplo de la Figura 1, el dispositivo 12 de origen incluye una fuente 20 de vídeo, una unidad 21 de procesamiento de profundidad, un codificador 22 de vídeo, y una interfaz 24 de salida. El dispositivo 16 de destino incluye una interfaz 26 de entrada, un decodificador 28 de vídeo, y un dispositivo 30 de visualización. De acuerdo con la presente divulgación, el codificador 22 de vídeo del dispositivo 12 de origen se puede configurar para aplicar una o más de las técnicas de la presente divulgación como parte de un procedimiento de codificación de vídeos. Del mismo modo, el decodificador 28 de vídeo del dispositivo 16 de destino se puede configurar para aplicar una o más de las técnicas de la presente divulgación como parte de un procedimiento de decodificación de vídeos.

El codificador 22 de vídeo puede aplicar también procesos de transformación, cuantificación, y codificación por entropía para reducir aún más la tasa de bits asociada con la comunicación de los bloques residuales. Las técnicas de transformación pueden comprender transformaciones de coseno discretas (DCT) o procedimientos conceptualmente similares. Como alternativa, se pueden utilizar transformaciones Wavelet, transformaciones enteras, u otros tipos de transformaciones. En un procedimiento de DCT, como un ejemplo, un conjunto de valores de píxeles se convierten en coeficientes de transformación, lo que representa la energía de los valores de píxel en el dominio de frecuencia. El codificador 22 de vídeo puede cuantificar también los coeficientes de transformación, que pueden generalmente implicar un procedimiento que reduce el número de bits asociados con el correspondiente coeficiente de transformación. La codificación por entropía puede incluir uno o más procedimientos que comprimen colectivamente los datos para su salida a una cadena de bits, donde los datos comprimidos pueden incluir, por ejemplo, una secuencia de modos de codificación, información de movimiento, patrones de bloque codificados, y coeficientes de transformaciones cuantificados. Ejemplos de codificación por entropía incluyen, pero no se limitan a, la codificación de longitud variable adaptativa contextual (CAVLC) y codificación aritmética binaria adaptativa contextual (CABAC).

Un bloque de vídeo codificado se puede representar mediante la información de predicción que se puede utilizar para crear o identificar un bloque de predicción, y un bloque residual de datos que se puede aplicar al bloque de predicción para recrear el bloque original. La información de predicción puede comprender uno o más vectores de movimiento que se utilizan para identificar el bloque de predicción de los datos. Mediante la utilización de los vectores de movimiento, el decodificador 28 de vídeo puede ser capaz de reconstruir los bloques de predicción que se utilizaron para codificar los bloques residuales. Por lo tanto, dado un conjunto de bloques residuales y un conjunto de vectores de movimiento (y posiblemente alguna sintaxis adicional), el decodificador 28 de vídeo puede reconstruir un fotograma de vídeo que fue codificado originalmente. La inter-codificación basándose en la estimación de movimiento y a la compensación de movimiento puede alcanzar cantidades relativamente altas de compresión sin pérdida de datos excesiva, ya que los fotogramas de vídeo sucesivos u otros tipos de unidades codificadas son a menudo similares. Una secuencia de vídeo codificada puede comprender bloques de datos residuales, vectores de movimiento (cuando se codifican por inter-predicción), indicaciones de modos de intra-predicción para intra-predicción, y elementos de sintaxis.

El codificador 22 de vídeo puede utilizar también técnicas intra-predicción para codificar bloques de vídeo relativos a los bloques de vídeo de un fotograma o porción común vecina. De esta manera, el codificador 22 de vídeo predice espacialmente los bloques. El codificador 22 de vídeo se puede configurar con una variedad de modos de intra-predicción, que generalmente corresponden a varias direcciones de predicción espacial. Al igual que con la estimación de movimiento, el codificador 22 de vídeo se puede configurar para seleccionar un modo de intra-

predicción basándose en un componente de luminancia de un bloque, a continuación, volver a utilizar el modo de intra-predicción para codificar los componentes de crominancia del bloque. Además, de acuerdo con las técnicas de la presente divulgación, el codificador 22 de vídeo puede volver a utilizar el modo de intra-predicción para codificar un componente de profundidad del bloque.

- 5 Mediante la reutilización de la información de movimiento y del modo de intra-predicción para codificar un componente profundidad de un bloque, estas técnicas pueden simplificar el procedimiento de mapas de profundidad de codificación. Por otra parte, las técnicas descritas en la presente memoria pueden mejorar la eficacia de la cadena de bits. Es decir, la cadena de bits solo tiene que indicar algunos elementos de sintaxis una vez en la cabecera de porciones para los componentes de vista de textura, en lugar de señalar el elemento de sintaxis
10 adicional en una cabecera de porciones de para una porción de componentes de vista de profundidad.

Opcionalmente, un componente de vista textura también puede reutilizar su correspondiente componente de vista de profundidad de la misma manera.

- Una vez más, el sistema 10 ilustrado de la Figura 1 no es más que un ejemplo. Las diversas técnicas de la presente divulgación se pueden realizar por cualquier dispositivo de codificación que soporte la codificación predictiva basándose en bloques, o por cualquier dispositivo de decodificación que soporte la decodificación predictiva basándose en bloques. El dispositivo 12 de origen y el dispositivo 16 de destino son meramente ejemplos de tales dispositivos de codificación, en el que dispositivo 12 de origen genera datos de vídeo codificados para su transmisión al dispositivo 16 de destino. En algunos casos, los dispositivos 12 y 16 pueden operar de manera sustancialmente simétrica, de manera que cada uno de los dispositivos 12 y 16 incluye componentes de codificación y descodificación de vídeos. Por lo tanto, el sistema 10 puede soportar la transmisión de vídeo unidireccional o bidireccional entre los dispositivos 12 y 16 de vídeo, por ejemplo, para streaming de vídeo, reproducción de vídeo, difusión del vídeo, o la videotelefonía.

- La fuente 20 de vídeo del dispositivo 12 de origen incluye un dispositivo de captura de vídeo, como una cámara de vídeo, un archivo de vídeo que contiene el vídeo capturado previamente, o un canal de vídeo de un proveedor de contenido de vídeo. Como alternativa, la fuente 20 de vídeo puede generar datos basándose en gráficos computarizados como la fuente de vídeo, o una combinación de vídeo en vivo, vídeo archivado, y/o vídeos generados por ordenador. En algunos casos, si la fuente 20 de vídeo es una cámara de vídeo, el dispositivo 12 de origen y el dispositivo 16 de destino pueden formar los denominados teléfonos con cámara o videoteléfonos u otros dispositivos móviles configurados para manipular datos de vídeo, tales como dispositivos informáticos de tipo tableta. En cada caso, el vídeo capturado, pre-capturado o generado por ordenador se puede codificar por el codificador 22 de vídeo. La fuente 20 de vídeo captura una vista y la proporciona a la unidad de procesamiento 21 de profundidad.

- La fuente 20 de vídeo proporciona una vista 2 a la unidad 21 de procesamiento de profundidad para el cálculo de la imagen de la profundidad de los objetos en la vista 2. En algunos ejemplos, la vista 2 comprende más de una vista. Una imagen de la profundidad se determina para los objetos en la vista 2 capturados por la fuente 20 de vídeo. La unidad 21 de procesamiento de profundidad se configura para calcular automáticamente valores de profundidad de los objetos en la imagen de la vista 2. Por ejemplo, la unidad 21 de procesamiento de profundidad calcula valores de profundidad para los objetos en función de información de luminancia. En algunos ejemplos, la unidad 21 de procesamiento de profundidad se configura para recibir la información de profundidad de un usuario. En algunos ejemplos, la fuente 20 de vídeo captura dos vistas de una escena a diferentes perspectivas y, a continuación, calcula la información de profundidad para los objetos de la escena basándose en la disparidad entre los objetos en las dos vistas. En diversos ejemplos, la fuente 20 de vídeo comprende una cámara bidimensional estándar, un sistema de dos cámaras que proporciona una vista estereoscópica de una escena, una matriz de cámara que captura múltiples vistas de la escena, o una cámara que captura una vista más información de profundidad.

- La unidad 21 de procesamiento de profundidad proporciona componentes de vista de textura 4 y componentes 6 de vista de profundidad al codificador 22 de vídeo. La 21 unidad de procesamiento de profundidad puede también ofrecer vistas 2 directamente al codificador 22 de vídeo. La información 6 de profundidad comprende una imagen del mapa de profundidad para la vista 2. Una imagen del mapa de profundidad puede comprender un mapa de valores de profundidad para cada región de píxeles asociados con un área (por ejemplo, bloque, porción, o fotograma) a mostrar. Una región de píxeles incluye un único píxel o un grupo de uno o más píxeles. Algunos ejemplos de los mapas de profundidad tienen un componente de profundidad por píxel. En otros ejemplos, hay múltiples componentes de profundidad por píxel. Los mapas de profundidad se pueden codificar una manera sustancialmente similar a los datos de textura, por ejemplo, mediante intra-predicción o inter-predicción en relación con otros datos de profundidad, previamente codificados. En otros ejemplos, los mapas de profundidad se codifican de manera diferente a como se codifican los datos de textura.

- El mapa de profundidad se puede estimar en algunos ejemplos. Cuando hay más de una vista presente, la coincidencia estereo se puede utilizar para estimar mapas de profundidad. Sin embargo, en la conversión de 2D a 3D, puede ser más difícil estimar la profundidad. Sin embargo, el mapa de profundidad estimado por diferentes procedimientos se puede utilizar para la representación 3D basándose en representación basándose en la profundidad de la imagen (DIBR).

Aunque la fuente 20 de vídeo puede proporcionar múltiples vistas de una escena y la unidad 21 de procesamiento de profundidad puede calcular la información de profundidad a partir de las múltiples vistas, el dispositivo 12 de origen puede transmitir una vista general, además de la información de profundidad para cada vista de una escena.

5 Cuando la vista 2 es una imagen fija digital, el codificador 22 de vídeo se puede configurar para codificar la vista 2 como, por ejemplo, una imagen de grupo Mixto de Expertos en Fotografía (JPEG). Cuando la vista 2 es una porción de datos de vídeo, el codificador 22 de vídeo se configura para codificar la primera vista 50 de acuerdo con un estándar de codificación de vídeos tal como, por ejemplo Grupo De Expertos En Imágenes En Movimiento (MPEG), la Organización Internacional de Normalización (ISO)/Comisión Electrotécnica Internacional (IEC) MPEG-1 Visual, ISO/IEC MPEG-2 Visual, ISO/IEC MPEG-4 Visual, la Unión Internacional de Telecomunicaciones (UIT) H.261, UIT-T 10 H.262, UIT-T H.263, ITU-T H.264/MPEG-4, H.264 Codificación de Vídeo Avanzada (AVC), el próximo estándar de Codificación Avanzada De Vídeo De Alta Eficiencia (HEVC) (también conocido como H.265), u otros estándares de codificación de vídeos. El codificador 22 de vídeo puede incluir información 6 de profundidad junto con la imagen codificada para formar un bloque 8 codificado, que incluye datos de imagen codificados, junto con la información 6 de profundidad. El codificador 22 de vídeo pasa el bloque 8 codificado a la interfaz 24 de salida. El bloque 8 15 codificado se puede transferir a la interfaz 26 de entrada en una cadena de bits que incluye información de señalación, junto con el bloque 8 codificado a través del enlace 15.

La información de vídeo codificada incluye componentes 4 de textura e información 6 de profundidad. Los componentes 4 de textura pueden incluir componentes de luminancia (luma) y de crominancia (croma) de la información del vídeo. Los componentes de luma describen generalmente el brillo, mientras que los componentes de crominancia describen generalmente matices de color. La unidad 21 de procesamiento de profundidad extrae la información 21 de profundidad de un mapa de profundidad de la vista 2. El codificador 22 de vídeo puede codificar los componentes 4 de vista de textura y los componentes 6 de vista de profundidad en un solo bloque 8 codificado de datos de vídeo codificados. Del mismo modo, el codificador 22 de vídeo puede codificar el bloque de tal manera que el movimiento o información de modo de intra-predicción para el componente de luminancia se reutiliza para los componentes de cromina y el componente de profundidad. Los elementos de sintaxis utilizados para los componentes de vista de textura se pueden utilizar para predecir elementos de sintaxis similares para los componentes de vista 20 profundidad.

En algunos ejemplos, el componente de vista del mapa de profundidad no se puede codificar utilizando técnicas de predicción inter-vistas, incluso cuando el componente de vista de textura correspondiente se codifica utilizando técnicas de predicción inter-vistas. Por ejemplo, el componente de vista del mapa de profundidad se puede predecir utilizando la predicción intra-vistas cuando el componente de vista de textura correspondiente prevé el uso de predicción inter-vistas. Por ejemplo, la predicción inter-vistas de un componente de vista de textura predice la información de vista de textura a partir de datos de una vista diferente como la vista correspondiente al componente de vista de textura. Por el contrario, la información de vista de profundidad de predicción intra-vistas predice la información de profundidad a partir de datos de la misma vista como la vista correspondiente a la información de 30 vista de profundidad.

A pesar de utilizar diferentes técnicas de predicción, algunos elementos de sintaxis para el componente de vista del mapa de profundidad se pueden predecir a partir de los elementos de sintaxis correspondientes en la cabecera de porciones del componente de vista de textura correspondiente. Sin embargo, la información de la cabecera de porciones para el componente de vista del mapa de profundidad puede contener información relacionada con una construcción de la lista de imágenes de referencia. Es decir, la información relacionada con la construcción de la lista de imágenes referencia se puede señalar en la cabecera de porciones para el componente de vista del mapa de profundidad. Por ejemplo, un número de imágenes de referencia que se utilizan y una indicación de que las imágenes de referencia se utilizan para predecir el componente de vista del mapa de profundidad se puede señalar en la cabecera de porciones para el componente de mapa de profundidad vista. Información similar se puede 40 señalar también en una cabecera de porciones para el componente de vista de textura correspondiente.

En algunos ejemplos, el dispositivo 12 de origen incluye un módem que modula el bloque 8 codificado de acuerdo con un estándar de comunicación, por ejemplo, como Acceso Múltiple Por División De Código (CDMA) u otro estándar. Un módem puede incluir diversos mezcladores, filtros, amplificadores u otros componentes diseñados para la modulación de la señal. La interfaz 24 de salida puede incluir circuitos diseñados para la transmisión de datos, incluyendo los amplificadores, filtros, y una o más antenas. El bloque 8 codificado se transmite al dispositivo 14 de destino a través de la interfaz 24 de salida y el enlace 15. En algunos ejemplos, en lugar de transmitirse a través de un canal de comunicación, el dispositivo 12 de origen almacena los datos de vídeo codificados, incluidos los bloques que tienen componentes de textura y profundidad, en un dispositivo 32 de almacenamiento, tal como un disco de vídeo digital (DVD), disco Blu-ray, una unidad flash o similares. 50 55

La interfaz 26 de entrada del dispositivo 14 de destino recibe información a través del enlace 15. En algunos ejemplos, el dispositivo 14 de destino incluye un módem que desmodula la información. Como la interfaz 24 de salida, la interfaz 26 de entrada puede incluir circuitos diseñados para recibir datos, incluyendo amplificadores, filtros, y una o más antenas. En algunos casos, la interfaz 24 de salida y/o interfaz 26 de entrada se pueden incorporar dentro de un único componente transceptor que incluye tanto circuitos de recepción como de transmisión. Un módem puede incluir diversos mezcladores, filtros, amplificadores u otros componentes diseñados para la 60

demodulación de la señal. En algunos casos, un módem puede incluir componentes para realizar tanto la modulación como la demodulación.

Una vez más, el procedimiento de codificación de vídeos realizado por el codificador 22 de vídeo puede implementar una o más de las técnicas descritas en la presente memoria durante la codificación de inter-predicción, que pueden incluir la estimación de movimiento y la compensación de movimiento, y la codificación de intra-predicción. El procedimiento de decodificación de vídeos realizado por decodificador 28 de vídeo puede realizar también tales técnicas durante una etapa de compensación de movimiento del procedimiento de decodificación.

El término "codificador" se utiliza aquí para referirse a un dispositivo o aparato informático especializado que realiza la codificación de vídeos o la decodificación de vídeos. El término "codificador" se refiere generalmente a cualquier codificador de vídeo, decodificador de vídeo, o codificador/decodificador combinados (codec). El término "codificación" se refiere a la codificación o decodificación. La expresión "bloques codificados", "unidad de bloque codificado," o "unidad codificada" se puede referir a cualquier unidad decodificable independiente de un fotograma de vídeo como un fotograma completo, una porción de un fotograma, un bloque de datos de vídeo, o u otra unidad independientemente decodificable definida de acuerdo con las técnicas de codificación utilizadas.

El dispositivo 30 de visualización muestra los datos de vídeo decodificados a un usuario, y puede comprender cualquiera de una variedad de uno o más dispositivos de visualización tal como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, un diodo emisor de luz orgánica (OLED), u otro tipo de dispositivo de visualización. En algunos ejemplos, el dispositivo 30 de visualización corresponde a un dispositivo capaz de la reproducción tridimensional. Por ejemplo, el dispositivo 30 de visualización puede comprender una pantalla estereoscópica, que se utiliza junto con gafas utilizadas por un espectador. Las gafas pueden comprender gafas activas, en cuyo caso el dispositivo 30 de visualización alterna rápidamente entre imágenes de diferentes vistas de forma sincronizada con cerramiento alternativo de las lentes de gafas activas. Como alternativa, las gafas pueden comprender gafas pasivas, en cuyo caso el dispositivo 30 de visualización muestra imágenes de diferentes vistas de forma simultánea, y las gafas pasivas pueden incluir lentes polarizadas que generalmente se polarizan en direcciones ortogonales para filtrar entre las diferentes vistas.

En el ejemplo de la Figura 1, el enlace 15 puede comprender cualquier medio de comunicación inalámbrica o por cable, tales como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión físicas, o cualquier combinación de medios de comunicación inalámbrica y por cable. El enlace 15 puede formar parte de una red basándose en paquetes, tal como una red de área local, una red de área amplia, o una red global tal como Internet. El enlace 15 representa generalmente cualquier medio de comunicación adecuado, o una colección de diferentes medios de comunicación, para transmitir datos de vídeo del dispositivo 12 de origen al dispositivo 14 de destino. El enlace 15 puede incluir enrutadores, conmutadores, estaciones base, o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo 12 de origen al dispositivo 14 de destino.

El codificador 22 de vídeo y decodificador 28 de vídeo pueden operar de acuerdo con un estándar de compresión de vídeo, tales como el estándar UIT-T H.264, que se describe alternativamente como MPEG-4 Parte 10, Codificación de Vídeo Avanzada (AVC). Los estándares de compresión de vídeo adicionales que se basan en el estándar ITU H.264/AVC que se pueden utilizar por el codificador 22 de vídeo y el decodificador 28 de vídeo incluyen el estándar de codificación de vídeos escalable (SVC), que es una extensión escalable al estándar UIT H.264/AVC. Otro estándar bajo el que el codificador 22 de vídeo y decodificador 28 de vídeo puede operar incluye el estándar de codificación de vídeos de múltiples vistas (MVC), que es una extensión de múltiples vistas del estándar ITU H.264/AVC. Las técnicas de la presente divulgación, sin embargo, no se limitan a cualquier estándar de codificación de vídeos en particular.

En algunos aspectos, el codificador 22 de vídeo y el decodificador 28 de vídeo pueden cada uno estar integrados con un codificador y decodificador de audio, y pueden incluir unidades apropiadas MUX-DEMUX u otro hardware y software, para manejar la codificación de audio y vídeo en un flujo de datos común o en flujos de datos independientes. Si corresponde, las unidades MUX-DEMUX pueden cumplir con el protocolo multiplexor ITU H.223, u otros protocolos tales como el protocolo de datagramas de usuario (UDP).

Cada uno del codificador 22 de vídeo y del decodificador 28 de vídeo se puede implementar como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados de aplicación específica (ASIC), matrices de compuertas programables (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando cualquiera o todas las técnicas de la presente divulgación se implementan en software, un dispositivo de aplicación podrá incluir además hardware para almacenar y/o ejecutar las instrucciones del software, por ejemplo, una memoria para almacenar las instrucciones y una o más unidades de procesamiento para ejecutar las instrucciones. Cada uno del codificador 22 de vídeo y del decodificador 28 de vídeo se pueden incluir en uno o más codificadores o decodificadores, cualquiera de los que se puede integrar como parte de un códec combinado que proporciona capacidades de codificación y decodificación en un dispositivo móvil, dispositivo suscriptor, dispositivo de difusión, servidor, o similares respectivos.

Una secuencia de vídeo incluye típicamente una serie de fotogramas de vídeo, también denominados como imágenes de vídeo. El codificador 22 de vídeo opera sobre bloques de vídeo dentro de fotogramas de vídeo

individuales con el fin de codificar los datos de vídeo. Los bloques de vídeo pueden tener tamaños fijos o variables, y pueden diferir en tamaño de acuerdo con un estándar de codificación específico. Cada fotograma de vídeo incluye una serie de uno o más sectores. En el estándar ITU-T H.264, por ejemplo, cada porción incluye una serie de macrobloques, que pueden estar dispuestos en sub-bloques. El estándar H.264 compatible con la intra-predicción en diversos tamaños de bloque para la codificación de vídeos bidimensional (2D), tales como 16 por 16, 8 por 8, o 4 por 4 para componentes luma, y 8 por 8 para componentes de croma, así como la inter-predicción en diversos tamaños de bloque, tales como 16 por 16, 16 por 8, 8 por 16, 8 por 8, 8 por 4, 4 por 8 y 4 por 4 para componentes de luminancia y los tamaños a escala correspondientes de los componentes de croma. Los bloques de vídeo pueden comprender bloques de datos de píxeles, o bloques de coeficientes de transformación, por ejemplo, después de un procedimiento de transformación, tales como la transformación de coseno discreta (DCT) o un procedimiento de transformación conceptualmente similar. Estas técnicas pueden extenderse a vídeos en 3D.

Los bloques de vídeo más pequeños pueden proporcionar una mejor resolución, y se pueden utilizar para ubicaciones de un fotograma de vídeo que incluye altos niveles de detalle. En general, los macrobloques y los diversos sub-bloques pueden considerarse como bloques de vídeo. Además, una porción se puede considerar como una serie de bloques de vídeo, como macrobloques y/o sub-bloques. Cada porción puede ser una unidad independiente decodificable de un fotograma de vídeo. Como alternativa, los propios fotogramas pueden ser unidades descifrables, u otras partes de un fotograma se pueden definir como unidades descifrables.

Los macrobloques en 2D de la norma ITU-T H.264 se pueden extender a 3D mediante codificación de la información de profundidad de un mapa de profundidad o mapa de paralaje, junto con los componentes de luma y croma asociados (es decir, los componentes de textura) para ese fotograma o porción de vídeo. El mapeo Parallax (también conocido como mapeo de desplazamiento virtual o mapeo de desfase) desplaza componentes de vista de textura en una ubicación de píxel basándose en una función de un ángulo de vista y un mapa de altura en la ubicación del píxel. El codificador 22 de vídeo puede codificar la información de profundidad como un vídeo monocromático.

Para codificar los bloques de vídeo, como un bloque codificado, El codificador 22 de vídeo realiza intra o inter-predicción para generar uno o más bloques de predicción. El codificador 22 de vídeo resta los bloques de predicción de los bloques de vídeo originales a codificarse para generar bloques residuales. Por lo tanto, los bloques residuales pueden representar diferencias píxel por píxel entre los bloques que se codifican y los bloques de predicción. El codificador 22 de vídeo puede realizar una transformación en los bloques residuales para generar bloques de coeficientes de transformación. Siguiendo las técnicas de codificación y transformación predictivas basándose en intra o inter, el codificador 22 de vídeo puede cuantificar los coeficientes de transformación. La cuantificación se refiere generalmente a un procedimiento en el que los coeficientes se cuantifican con la posibilidad de reducir la cantidad de datos utilizados para representar los coeficientes. Después de la cuantificación, se puede realizar la codificación por entropía de acuerdo con una metodología de codificación por entropía, como la codificación de longitud variable adaptativa contextual (CAVLC) o la codificación aritmética binaria adaptativa contextual (CABAC). Los detalles adicionales de un procedimiento de codificación realizado por el codificador 22 de vídeo se describen a continuación con respecto a la Figura 2.

Actualmente se están realizando esfuerzos para desarrollar un nuevo estándar de codificación de vídeos, actualmente denominado Codificación de vídeos de Alta Eficiencia (HEVC). El próximo estándar se conoce también como H.265. Los esfuerzos de normalización se basan en un modelo de un dispositivo de codificación de vídeos referido como el Modelo de prueba HEVC (HM). El HM presume de varias capacidades de los dispositivos de codificación de vídeos con respecto a los dispositivos de acuerdo con, por ejemplo, e UIT-T H.264/AVC. Por ejemplo, si bien H.264 proporciona nueve modos de codificación intra-predicción, el HM ofrece un máximo de treinta y tres modos de codificación intra-predicción. La HEVC se puede extenderse para soportar técnicas de información de cabecera de porciones como se describe en la presente memoria.

El HM se refiere a un bloque de datos de vídeo como una unidad de codificación (CU). Los datos de sintaxis dentro de una cadena de bits pueden definir una unidad de codificación más grande (LCU), que es una unidad de codificación más grande en términos del número de píxeles. En general, una CU tiene un propósito similar a un macrobloque de H.264, excepto que una CU no tiene una distinción tamaño. Un bloque codificado puede ser una CU según la norma HM. Por tanto, una CU se puede dividir en sub-UC. En general, las referencias en esta revelación a una CU se pueden referir a una unidad de codificación más grande (LCU) de una imagen o una sub-CU de una LCU. Una LCU se puede dividir en una sub-UC, y cada sub-CU se puede dividir en una sub-UC. Los datos de sintaxis para una cadena de bits pueden definir el número máximo de veces en el que se puede dividir una LCU, lo que se refiere como la profundidad de CU. Por consiguiente, una cadena de bits puede definir también una unidad de codificación más pequeña (SCU). La presente divulgación utiliza también el término "bloque" para referirse a cualquiera de una CU, unidad de predicción (PU), o unidad de transformación (TU).

Una LCU se puede asociar con una estructura de datos en árbol cuádruple. En general, una estructura de datos en árbol cuádruple incluye un nodo por cada CU, donde un nodo raíz corresponde a la LCU. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hojas, cada uno de los que corresponde a una de las sub-UC. Cada nodo de la estructura de datos en árbol cuádruple puede proporcionar datos de sintaxis para la CU correspondiente. Por ejemplo, un nodo en el árbol cuádruple puede incluir un indicador dividido, que indica si la

CU correspondiente al nodo se divide en sub-UC. Los elementos de sintaxis de una CU se pueden definir de forma recursiva, y pueden depender de si la CU se divide en sub-UC.

Una CU que no está dividida puede incluir una o más unidades de predicción (PU). En general, una PU representa todo o una parte de la CU correspondiente, e incluye datos para recuperar una muestra de referencia para la PU. Por ejemplo, cuando la PU se codifica en intra-modo, la PU puede incluir datos que describen un modo de intra-predicción para la PU. Como otro ejemplo, cuando la PU se codifica en inter-modo, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, la precisión de píxel de un cuarto o un octavo de precisión de píxel), un fotograma de referencia al que apunta el vector de movimiento, y/o una lista de referencia (por ejemplo, la lista 0 o lista 1) para el vector de movimiento. El vector de movimiento se puede tratar también como teniendo diferentes resoluciones para componentes de vista de textura y componentes de vista de profundidad. Los datos para la CU que define las PU puede describir también, por ejemplo, la partición de la CU en una o más PU. Los modos de particiones pueden ser diferentes si la CU no se codifica, se codifica en el modo intra-predicción, o se codifica el modo de inter-predicción.

Una CU que tiene una o más PU puede incluir también una o más unidades de transformación (TU). Después de la predicción utilizando una PU, el codificador 22 de vídeo puede calcular un valor residual para la parte de la CU correspondiente a la PU. El valor residual se puede transformar, registrar y cuantificar. Una TU no está necesariamente limitada al tamaño de una PU. Por lo tanto, la TU puede ser mayor o menor que las PU correspondiente para la misma CU. En algunos ejemplos, el tamaño máximo de una TU puede corresponder al tamaño de la CU correspondiente.

Como se ha señalado anteriormente, la intra-predicción incluye la predicción de una PU de una CU actual de una imagen desde las CU previamente codificadas de la misma imagen. Más específicamente, el codificador 22 de vídeo puede intra-predicir una CU actual de una imagen utilizando un modo de intra-predicción particular. Un codificador HM se puede configurar con hasta treinta y tres modos de intra-predicción. Por lo tanto, para soportar un mapeo de uno a uno entre los modos de intra-predicción direccionales y las transformaciones direccionales, los codificadores y decodificadores del HM tendrían que almacenar 66 matrices para cada una tamaño de transformación soportado. Además, los tamaños de bloque para los todos los treinta y tres modos de intra-predicción se soportan pueden ser bloques relativamente grandes, por ejemplo, píxeles 32x32, 64x64 píxeles, o incluso más grandes.

En el dispositivo 16 de destino, el decodificador 28 de vídeo recibe los datos 8 de vídeo codificados. El decodificador 28 de vídeo decodifica por entropía los datos 8 de vídeo codificados recibidos, tales como un bloque codificado, de acuerdo con una metodología de codificación por entropía, como CABAC o CAVLC, para obtener los coeficientes cuantificados. El decodificador 28 de vídeo aplica funciones de cuantificación inversa (de-cuantificación) y de transformación inversa para reconstruir el bloque residual en el dominio de píxeles. El decodificador 28 de vídeo genera también un bloque de predicción basándose en información de control o información de sintaxis (por ejemplo, modo de codificación, vectores de movimiento, la sintaxis que definen los coeficientes de filtro y similares) incluida en los datos de vídeo codificados. El decodificador 28 de vídeo calcula una suma del bloque de predicción y el bloque residual reconstruido para producir un bloque de vídeo reconstruido para su visualización. Los detalles adicionales de un procedimiento de decodificación ejemplar realizado por el decodificador 28 de vídeo se describen a continuación con respecto a la Figura 5.

Como se describe en la presente memoria, Y puede representar de luminancia, Cb y Cr pueden representar dos valores diferentes de crominancia de un espacio de color YCbCr tridimensional (por ejemplo, matices azules y rojizos), y D puede representar la información de profundidad. En algunos ejemplos, cada ubicación de píxel puede definir, en realidad, tres valores de píxel para un espacio de color tridimensional y el valor de un píxel para la profundidad de la ubicación de píxel. En otros ejemplos, pueden haber diferentes números de componentes de luma por componente de croma. Por ejemplo, puede haber cuatro componentes luma por componente de croma. Además, los componentes de profundidad y textura pueden tener diferentes resoluciones. En tal ejemplo, puede no haber una relación de uno a uno entre los componentes de vista de textura (por ejemplo, componentes de luma) y los componentes de vista de profundidad. Las técnicas de la presente divulgación, sin embargo, pueden referirse a la predicción con respecto a una dimensión para fines de simplicidad. En la medida en que las técnicas se describen con respecto a los valores de píxeles en una dimensión, técnicas similares se pueden extender a las otras dimensiones. En particular, de acuerdo con un aspecto de la presente divulgación, el codificador 22 de vídeo y/o el decodificador 28 de vídeo pueden obtener un bloque de píxeles, en el que el bloque de píxeles incluye componentes de vista de textura y componentes de vista de profundidad.

En algunos ejemplos, el codificador 22 de vídeo y el decodificador 28 de vídeo pueden utilizar una o más técnicas de filtrado por interpolación durante la compensación de movimiento. Es decir, el codificador 22 de vídeo y/o el decodificador 28 de vídeo pueden aplicar un filtro de interpolación para filtrar el soporte que comprende conjuntos de posiciones de píxeles enteros llenas.

El decodificador 28 de vídeo del dispositivo 16 de destino recibe uno o más bloques codificados como parte de una cadena de bits de vídeo codificada junto con la información adicional, incluyendo elementos de sintaxis relativos a

los componentes de vista de textura. El decodificador 28 de vídeo puede representar datos de vídeo para la reproducción en 3D basándose en los bloques 8 codificados y elementos sintaxis. De acuerdo con las técnicas de la presente divulgación, y como se describe en mayor detalle a continuación, los elementos de sintaxis señalados para los componentes 4 de vista de textura se pueden utilizar para predecir elementos de sintaxis para los componentes 6 de vista de profundidad. Los elementos de sintaxis se pueden señalar en una cabecera de porciones para los componentes 4 de vista de textura. Los elementos de sintaxis correspondientes para componentes 6 de vista de profundidad se pueden determinar a partir de los elementos de sintaxis relacionados para los componentes 4 de vista de textura.

Algunos elementos de sintaxis para los componentes 6 de vista de profundidad se pueden señalar en una cabecera de porción para los componentes 6 de vista de profundidad, tal como una diferencia de parámetros de cuantificación entre el componente de mapa de profundidad y uno de los uno o más componentes de textura para una porción. El atributo también puede ser un indicador en el nivel de porción que indica si los parámetros de filtro de bucle utilizados para el componente de vista de profundidad son los mismos que los parámetros de filtro de bucle que se han señalado para los componentes de vista de textura. En otros ejemplos, los elementos de sintaxis se pueden señalar en el plano secuencia (por ejemplo, en una estructura de datos de un conjunto de parámetros de secuencia (SPS)), el nivel de imagen (por ejemplo, en una estructura de datos o cabecera de porciones de un conjunto de parámetros de imagen (PPS)), o el nivel de bloque (por ejemplo, en una cabecera de bloques), además del nivel de porción (por ejemplo, en una cabecera de porciones).

La Figura 2 es un diagrama de bloques que ilustra un ejemplo del codificador 22 de vídeo de la Figura 1 con más detalle. El codificador 22 de vídeo codifica unidades de bloque que señalan elementos de sintaxis para los componentes de vista de textura que se pueden utilizar para predecir los elementos de sintaxis para los componentes de vista de profundidad, de acuerdo con las técnicas de la presente divulgación. El codificador 22 de vídeo es un ejemplo de un dispositivo o aparato informático de vídeo especializado denominado aquí como "codificador". Como se muestra en la Figura 2, el codificador 22 de vídeo corresponde al codificador 22 de vídeo de un dispositivo 12 de origen. Sin embargo, en otros ejemplos, el codificador 22 de vídeo puede corresponder a un dispositivo diferente. En otros ejemplos, otras unidades (tales como, por ejemplo, otro codificador/decodificador (CODECS)) pueden también realizar técnicas similares a las realizadas por el codificador 22 de vídeo.

El codificador 22 de vídeo puede realizar al menos una de intra e inter-codificación de bloques dentro de fotogramas de vídeo, aunque los componentes de intra-codificación no se muestran en la Figura 2 para facilitar la ilustración. La intra-codificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de un fotograma de vídeo dado. La inter-codificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en vídeo dentro de los fotogramas adyacentes de una secuencia de vídeo. El intra-modo (modo I) se puede referir al modo de compresión basándose en el espacio. Los inter-modos como una predicción (modo P) o uno bidireccional (modo B) se pueden referirse a los modos de compresión basándose en el tiempo. Las técnicas de la presente divulgación se aplican durante la inter-codificación e intra-codificación. Sin embargo, por simplicidad y facilidad de la ilustración, las unidades de intra-codificación como una unidad de predicción espacial no se ilustran en la Figura 2.

Como se muestra en la Figura 2, el codificador 22 de vídeo recibe un bloque de vídeo dentro de un fotograma de vídeo a codificar. En un ejemplo, el codificador 22 de vídeo recibe componentes 4 de vista de textura y componentes 6 de vista de profundidad. En otro ejemplo, el codificador de vídeo recibe la vista 2 de la fuente 20 de vídeo.

En el ejemplo de la Figura 2, el codificador 22 de vídeo incluye una unidad 32 de procesamiento de predicción, unidad de codificación de predicción (MCU), unidad 33 de profundidad más vídeo en múltiples vistas de vídeo (MVD), la memoria 34, un primer sumador 48, una unidad 38 de procesamiento de transformación, una unidad 40 de cuantificación, y una unidad 46 de codificación por entropía. Para la reconstrucción bloques de vídeo, el codificador 22 de vídeo incluye también una unidad 42 de cuantificación inversa, una unidad 44 de procesamiento de transformación inversa, un segundo sumador 51, y una unidad 43 de desbloqueo. La unidad 43 de desbloqueo es un filtro de desbloqueo que filtra bloque límites para eliminar artefactos de autoarranque de vídeos reconstruidos. Si se incluye en el codificador 22 de vídeo, la unidad 43 de desbloqueo filtraría normalmente la salida del segundo sumador 51. La unidad 43 de desbloqueo puede determinar la información de desbloqueo para uno o más componentes de vista de textura. La unidad 43 de desbloqueo puede determinar también la información de desbloqueo para el componente de mapa de profundidad. En algunos ejemplos, la información de desbloqueo para el uno o más componentes de la textura puede ser diferente de la información de desbloqueo para el componente de mapa de profundidad. En un ejemplo, como se muestra en la Figura 2, la unidad 38 de procesamiento de transformación representa un bloque funcional, a diferencia de una "TU" en términos de HEVC.

La unidad 33 de profundidad más vídeo en múltiples vistas (MVD) recibe uno o más bloques de vídeo (con la etiqueta "BLOQUE DE VÍDEO" en la Figura 2) que comprenden componentes de textura e información de profundidad, como la componentes 4 de vista de textura y componentes 6 de vista de profundidad. La unidad 33 de MVD ofrece funcionalidad para que el codificador 22 de vídeo codifique los componentes de profundidad en una unidad de bloque. La unidad 33 de MVD proporciona los componentes de vista de textura y componentes de vista de profundidad, ya sea combinados o por separado, a la unidad 32 de procedimiento de predicción en un formato que permite que la unidad 32 de procedimiento de predicción procese la información de profundidad. La Unidad 33 MVD

5 puede también señalar a la unidad 38 de procesamiento de transformación que los componentes de vista de profundidad se incluyen con el bloque de vídeo. En otros ejemplos, cada unidad del codificador 22 de vídeo, como la unidad 32 de procesamiento de predicción, la unidad 38 de procesamiento de transformación, la unidad 40 de cuantificación, la unidad 46 de codificación por entropía, etc., comprende una funcionalidad para procesar la información de profundidad además de los componentes de vista de textura.

10 En general, el codificador 22 de vídeo codifica la información de profundidad de manera similar a la información de crominancia, en que la unidad 37 de compensación de movimiento se configura para volver a utilizar los vectores de movimiento calculados para un componente de luminancia de un bloque durante el cálculo de un valor predicho para un componente de profundidad del mismo bloque. Del mismo modo, una unidad de intra-predicción del codificador 22 de vídeo se puede configurar para utilizar un modo de intra-predicción seleccionado para el componente de luminancia (es decir, basándose en el análisis del componente de luminancia) cuando se codifica el componente de vista profundidad utilizando intra-predicción.

15 La unidad 32 de procesamiento de predicción incluye una unidad 35 de estimación de movimiento (ME) y una unidad 37 de compensación de movimiento (MC). La unidad 32 de procesamiento de predicción predice la información de profundidad para ubicaciones de píxel, así como para los componentes de textura. Uno o más filtros 39 de interpolación (referidos aquí como el "filtro 39") se pueden incluir en la unidad 32 de procesamiento de predicción y pueden invocarse por una o ambas de la unidad 35 de ME y la unidad 37 de MC para realizar la interpolación como parte de la estimación de movimiento y/o la compensación de movimiento. El filtro 39 de interpolación puede representar realmente una pluralidad de diferentes filtros para facilitar numerosos tipos diferentes de interpolación y de filtrado de tipo interpolación. Por lo tanto, la unidad 32 de procedimiento de predicción puede incluir una pluralidad de filtros de interpolación o de tipos de interpolación.

20 Durante el procedimiento de codificación, el codificador 22 de vídeo recibe un bloque de vídeo a codificar (etiquetado como "BLOQUE DE VÍDEO" en la Figura 2), y la unidad 32 de procedimiento de predicción realiza la codificación por inter-predicción para generar un bloque de predicción (con la etiqueta "BLOQUE DE PREDICCIÓN" en la Figura 2). El bloque de predicción incluye tanto componentes de vista de textura como información de vista de profundidad. Específicamente, la unidad 35 de ME puede realizar la estimación de movimiento para identificar el bloque de predicción en la memoria 34, y la unidad 37 de MC puede realizar la compensación de movimiento para generar el bloque de predicción.

25 La estimación de movimiento se considera normalmente el procedimiento de generación de vectores de movimiento, que estiman el movimiento de los bloques de vídeo. Un vector de movimiento puede, por ejemplo, indicar el desplazamiento de un bloque de predicción dentro de un fotograma o referencia de predicción (u otra unidad codificada, por ejemplo, porción) con relación al bloque a codificar dentro del fotograma actual (u otra unidad codificada). El vector de movimiento puede tener precisión de píxel completamente entera o sub-entera. Por ejemplo, tanto una componente horizontal como una componente vertical del vector de movimiento pueden tener respectivos componentes enteros completos y componentes sub-enteros. El fotograma de referencia (o parte del fotograma) se puede situar antes o después del fotograma de vídeo (o parte del fotograma de vídeo) a la que el bloque de vídeo actual pertenece. La compensación de movimiento se considera típicamente el procedimiento de buscar o generar el bloque de predicción a partir de la memoria 34, lo que puede incluir la interpolación o, de otro modo, la generación de datos de predicción basándose en el vector de movimiento determinado por la estimación de movimiento.

30 La unidad 35 de ME calcula al menos un vector de movimiento para el bloque de vídeo a codificar mediante la comparación del bloque de vídeo con los bloques de referencia de uno o más fotogramas de referencia (por ejemplo, un fotograma anterior y/o posterior). Los datos para los fotogramas de referencia se pueden almacenar en la memoria 34. La unidad 35 ME puede realizar la estimación de movimiento con una precisión de píxel fraccionario, a veces referida como la estimación de movimiento de píxel fraccionado, pel fraccionado, sub-entera, o sub-píxel. En la estimación de movimiento de píxel fraccionado, la unidad 35 de ME calcula un vector de movimiento que indica el desplazamiento a una ubicación diferente de la ubicación de píxel entero. Por lo tanto, el vector de movimiento puede tener la precisión de píxel fraccionario, por ejemplo, la precisión de un medio de píxel, la precisión de un cuarto de píxel, la precisión de un octavo de píxel, u otras precisiones de píxeles fraccionarios. De esta manera, la estimación de movimiento de píxel fraccionario permite que la unidad 32 de procesamiento de predicción estime el movimiento con mayor precisión que las ubicaciones de píxel entero (o píxel completo) y, por lo tanto, la unidad de procesamiento 32 genera la predicción de un bloque de predicción más preciso. La estimación de movimiento de píxeles fraccionarios permite que la unidad 32 de procesamiento de predicción prediga la información de profundidad a una primera resolución y prediga los componentes textura a una segunda resolución. Por ejemplo, los componentes de textura se predicen con una precisión de píxel completo, mientras que la información de profundidad se predice con una precisión de medio píxel. En otros ejemplos, otras resoluciones del vector de movimiento se pueden utilizar para la información de profundidad y los componentes de textura.

35 La unidad 35 de ME puede invocar uno o más filtros 39 para las interpolaciones necesarias durante el procedimiento de estimación de movimiento. En algunos ejemplos, la memoria 34 puede almacenar valores interpolados para píxeles sub-enteros, que se pueden calcular, por ejemplo, con el sumador 51 utilizando los filtros 39. Por ejemplo, el sumador 51 puede aplicar filtros 39 a los bloques reconstruidos que van a almacenarse en la memoria 34.

Una vez que la unidad 32 de procesamiento de predicción ha generado el bloque de predicción, el codificador 22 de vídeo forma un bloque de vídeo residual (con la etiqueta "BLOQUE RESID." en la Figura 2) restando el bloque de predicción del bloque de vídeo original que se codifica. Esta resta puede ocurrir entre los componentes de textura en el bloque de vídeo original y los componentes de textura en el bloque de predicción, así como para la información de profundidad en el bloque de vídeo original o el mapa de profundidad de la información de profundidad en el bloque de predicción. El sumador 48 representa el componente o componentes que realizan esta operación de resta.

La unidad 38 de procesamiento de transformación aplica una transformación, tal como una transformación de coseno discreta (DCT) o una transformación conceptualmente similar, al bloque residual, produciendo un bloque de vídeo residual que comprende coeficientes de transformación de bloques. Se debe entender la unidad 38 de procesamiento de transformación representa el componente del codificador 22 de vídeo que aplica una transformación a los coeficientes residuales de un bloque de datos de vídeo, en contraste con una TU de una CU como se define por HEVC. La unidad 38 de procesamiento de transformación, por ejemplo, puede realizar otras transformaciones, tales como las definidas por el estándar H.264, que son conceptualmente similar a DCT. Tales transformaciones incluyen, por ejemplo, transformaciones direccionales (tales como el teorema de transformación Karhunen-Loeve), transformaciones wavelet, transformaciones enteras, transformaciones de sub-banda, u otros tipos de transformaciones. En cualquier caso, la unidad 38 de procesamiento de transformación aplica la transformación al bloque residual, produciendo un bloque de coeficientes residuales de transformación. La unidad 38 de procesamiento de transformación puede aplicar el mismo tipo de transformación tanto a los componentes de textura como a la información de profundidad en los bloques residuales correspondientes. Habrá bloques residuales separados para cada componente de textura y profundidad. La transformación convierte la información residual de un dominio de pixel a un dominio de frecuencia.

La unidad 40 de cuantificación cuantifica los coeficientes de transformación residuales para reducir aún más la tasa de bits. El procedimiento de cuantificación puede reducir la profundidad de bits asociado con algunos o todos de los coeficientes. La unidad 40 de cuantificación puede cuantificar un residuo de codificación de imagen de profundidad. Después de la cuantificación, la unidad 46 de codificación por entropía codifica por entropía los coeficientes de transformación cuantificados. Por ejemplo, la unidad 46 de codificación por entropía puede realizar CAVLC, CABAC, u otra metodología de codificación por entropía.

La unidad 46 de codificación por entropía puede codificar también uno o más vectores de movimiento y soportar la información obtenida de la unidad 32 de procesamiento de predicción u otro componente del codificador 22 de vídeo, tal como la unidad 40 de cuantificación. El uno o más elementos de sintaxis de predicción pueden incluir un modo de codificación, los datos de uno o más vectores de movimiento (por ejemplo, componentes horizontal y vertical, identificadores de la lista de referencia, índices de lista, y/o información de señalación de la resolución de vectores de movimiento), una indicación de una técnica de interpolación utilizada, un conjunto de coeficientes de filtro, una indicación de la resolución relativa de la imagen de profundidad con respecto a la resolución del componente de luma, una matriz de cuantificación para el residuo de codificación de imagen profundidad, la información de desbloqueo para la imagen de profundidad, o cualquier otra información asociada con la generación del bloque de predicción. Estos elementos de sintaxis de predicción se pueden proporcionar en el nivel de secuencia o en el nivel de imagen.

El uno o más elementos de sintaxis pueden incluir también una diferencia de parámetros de cuantificación (QP) entre el componente luma y el componente de profundidad. La diferencia de QP se puede señalar al nivel de porción y se puede incluir en una cabecera de porciones para los componentes de vista de textura. Otros elementos de sintaxis también se puede señalar a un nivel de la unidad de bloque codificado, incluyendo un patrón de bloque codificado para el componente de vista de profundidad, un delta QP para el componente de vista de profundidad, una diferencia de vectores de movimiento, u otra información asociada con la generación del bloque de predicción. La diferencia de vectores de movimiento se puede señalar como un valor delta entre un vector de movimiento de destino y un vector de movimiento de los componentes de textura, o como un valor delta entre el vector de movimiento de destino (es decir, el vector de movimiento del bloque que está siendo codificado) y una predictor de vectores de movimiento vecinos para el bloque (por ejemplo, una PU de una CU). Después de la codificación por entropía con la unidad 46 de codificación por entropía, los elementos de sintaxis y de vídeo codificados se pueden transmitir a otro dispositivo o archivar (por ejemplo, en la memoria 34) para su transmisión o posterior.

La unidad 42 de cuantificación inversa y la unidad 44 de procesamiento de transformación inversa aplican cuantificación inversa y transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. El bloque reconstruido residual (con la etiqueta "BLOQUE RESID. RECON." en la Figura 2) puede representar una versión reconstruida del bloque residual suministrado a la unidad 38 de procesamiento de transformación. El bloque residual reconstruido puede diferir del bloque residual generado en el sumador 48, debido a pérdida de detalle causada por las operaciones de cuantificación y de cuantificación inversa. El sumador 51 añade el bloque residual reconstruido al bloque de predicción compensado de movimiento producido por la unidad 32 de procesamiento de predicción para producir un bloque de vídeo reconstruido para su almacenamiento en la memoria 34. El bloque de vídeo reconstruido se puede utilizar por la unidad 32 de procesamiento de predicción como un bloque de referencia que se puede utilizar para codificar posteriormente una unidad de bloque en un fotograma de vídeo posterior o unidad codificada posterior.

De esta manera, el codificador 22 de vídeo representa un ejemplo de un codificador de vídeo configurado para recibir una unidad de bloque codificado que comprende un componente de vista indicativo de una vista de una imagen, en el que el componente de vista comprende uno o más componentes de vista de textura y un componente de vista de profundidad, genera una cabecera de porciones de textura para uno o más componentes de vista de textura, incluyendo elementos de sintaxis de textura, en el que los elementos de sintaxis de profundidad para el componente de vista de profundidad se pueden determinar a partir de los elementos de sintaxis de textura en la cabecera de porciones de textura.

En algunos casos, la información con respecto a la codificación de los componentes de vista de textura y de los componentes de vista de profundidad se indica como uno o más elementos de sintaxis para su inclusión en la cadena de bits codificados. En algunos ejemplos, una cabecera de porciones de profundidad comprende elementos de sintaxis que incluyen al menos uno de la ubicación del microbloque de partida, el tipo de porción, el parámetro de la imagen (PPS) que se utilizará, el delta QP entre el QP inicial de la porción y el QP que se ha señalado en el PPS, el orden de las imágenes de referencia (representado como `frame_num`), y una orden de visualización de la imagen actual (POC). La cabecera de porciones de profundidad puede comprender también al menos uno de una construcción de la lista de imágenes de referencia y los elementos de sintaxis relacionados, una operación de control de gestión de memoria y elementos de sintaxis relacionados, y una predicción ponderada y elementos de sintaxis relacionados.

La Figura 3 es un diagrama de un ejemplo de una estructura de predicción de MVC (MVC) para la codificación de vídeos en múltiples vistas. El MVC es una extensión de H.264/AVC. La estructura de predicción de MVC incluye tanto la predicción inter-imágenes dentro de cada vista como la predicción intra-imágenes. En la Figura 3, las predicciones se indican con flechas, donde el objeto apuntado utiliza el objeto apuntador para la referencia de predicción. La estructura de predicción de MVC de la Figura 3 se puede utilizar junto con una disposición de orden de decodificación primera en el tiempo. En un orden de decodificación primera en el tiempo, cada unidad de acceso se puede definir para contener imágenes codificadas de todas las vistas para un instante de tiempo de salida. El orden de decodificación de las unidades de acceso puede no ser idéntico al orden de salida o visualización.

En la MVC, la predicción inter-vistas se soporta por la compensación de movimiento de disparidad, que utiliza la sintaxis de la compensación de movimiento H.264/AVC, pero permite que una imagen en una vista diferente se coloque como imagen de referencia. La codificación de las dos vistas se podría soportar también por la MVC. Un codificador de MVC puede tomar más de dos vistas como una entrada de vídeo en 3D y un decodificador de MVC puede decodificar una representación de múltiples vistas. Un procesador con un decodificador de MVC puede decodificar contenido de vídeo en 3D con múltiples vistas.

Las imágenes en la misma unidad de acceso (es decir, con el mismo instante de tiempo) se pueden predecir inter-vistas en la MVC. Al codificar una imagen en una de las vistas no base, se puede añadir una imagen en una lista de imágenes de referencia si está en una vista diferente pero en un mismo instante de tiempo. Una imagen de referencia de la predicción inter-vistas se puede poner en cualquier posición de una lista de imágenes de referencia, al igual que cualquier imagen de referencia de la inter-predicción.

En la MVC, la predicción inter-vistas se puede considerar realizada como si el componente de vista en otra vista es una referencia de la inter-predicción. Las posibles referencias inter-vistas se pueden señalar en la extensión de MVC de un conjunto de parámetro de secuencia (SPS). Las posibles referencias inter-vistas se pueden modificar por el procedimiento de construcción de la lista de imágenes de referencia, lo que permite hacer posible un orden flexible de las referencias de inter-predicción o predicción inter-vistas.

En contraste, en el HEVC, la cabecera de porciones sigue un principio de diseño similar al del H.264/AVC. Además, una cabecera de porciones de HEVC puede contener una sintaxis de parámetro de filtro de bucle adaptativo (ALF) en la especificación de HEVC actual. En algunos ejemplos, la cabecera de porciones de profundidad comprende uno o más parámetros de filtro de bucle adaptativos.

En un codec 3DV, un componente de vista de cada vista en un instante de tiempo específico puede incluir un componente de vista de textura y un componente de vista de profundidad. Una estructura por porciones se puede utilizar para fines de capacidad de recuperación de error, es decir, para proporcionar resiliencia de error. Sin embargo, un componente de vista de profundidad solo puede ser significativo cuando se recibe correctamente el componente de vista de textura correspondiente. Al incluir todos los elementos de sintaxis para el componente de vista de profundidad, una cabecera de porciones de la unidad de NAL de un componente de vista de profundidad puede ser relativamente grande. El tamaño de la cabecera de porciones de profundidad se puede reducir mediante la predicción de algunos elementos de sintaxis a partir de los elementos de sintaxis en la cabecera de porciones de textura para los componentes de vista de textura.

Una cadena de bits se puede utilizar para transferir unidades de bloque de profundidad más vídeo en múltiples vistas y elementos de sintaxis entre, por ejemplo, el dispositivo 12 de origen y el dispositivo 16 de destino de la Figura 1. La cadena de bits puede cumplir con el estándar de codificación ITU H.264/AVC y, en particular, sigue una estructura de cadena de bits de codificación de vídeos en múltiples vistas (MVC). Es decir, en algunos ejemplos, la cadena de bits conforma la extensión MVC de H.264/AVC. En otros ejemplos, la cadena de bits conforma una

extensión en múltiples vistas de HEVC o la extensión en múltiples vistas de otro estándar. En aún otros ejemplos, se utilizan otros estándares de codificación.

Una disposición de orden (orden de decodificación) de la cadena de bits de MVC típica es una codificación de primera en el tiempo. Cada unidad de acceso se define para contener las imágenes codificadas de todas las vistas para un instante de tiempo de salida. El orden de decodificación de las unidades de acceso puede o no puede ser idéntico a la orden de salida o visualización. Por lo general, la predicción de MVC puede incluir tanto la predicción inter-imágenes dentro de cada vista como la predicción inter-vistas. En la MVC, la predicción inter-vistas se puede soportar por la compensación de movimiento de disparidad, que utiliza la sintaxis de la compensación de movimiento H.264/AVC, pero que permite que una imagen en una vista diferente se utilice como una imagen de referencia.

La codificación de dos vistas se soporta por la MVC. Una de las ventajas de MVC es que un codificador de MVC podría tomar más de dos vistas como una entrada de vídeo en 3D y un decodificador de MVC puede decodificar las dos vistas en una representación de múltiples vistas. Por lo tanto, un procesador con un decodificador de MVC puede tratar el contenido de vídeo en 3D como teniendo múltiples vistas. Anteriormente, la MVC no procesó la entrada del mapa de profundidad, similar a H.264/AVC con mensajes de información de mejora complementarios (SEI) (información estéreo o imágenes de entrelazado espaciales).

En el estándar H.264/AVC, se definen unidades de capa de abstracción de red (NAL) para proporcionar una representación de vídeo "soportada por la red" que abarque aplicaciones tales como la telefonía de vídeo, almacenamiento o streaming de vídeo. Las unidades de NAL se pueden clasificar de unidades de NAL de capa de codificación de vídeo (VCL) y unidades de NAL no-VCL. Las unidades de VCL pueden contener un motor de compresión de núcleo y comprenden el bloque, el macrobloque (MB), y los niveles de porciones. Otras unidades de NAL son unidades de NAL no VCL.

En un ejemplo de codificación de vídeos 2D, cada unidad de NAL contiene una cabecera de unidad de NAL de un byte y una carga útil de tamaño variable. Cinco bits se utilizan para especificar el tipo de unidad de NAL. Tres bits se utilizan para `nal_ref_idc`, lo que indica cuán importante es la unidad de NAL en términos de ser referenciada por otras imágenes (unidades de NAL). Por ejemplo, establecer `nal_ref_idc` igual a 0 significa que la unidad de NAL no se utiliza para la inter-predicción. Dado que el H.264/AVC se expande para incluir la codificación de vídeos en 3D, tal como el estándar de codificación de vídeos escalable (SVC), la cabecera de NAL puede ser similar a la del escenario en 2D. Por ejemplo, uno o más bits en la cabecera de la unidad de NAL se utilizan para identificar que la unidad de NAL es una unidad de NAL de cuatro componentes.

Las cabeceras de unidades de NAL se pueden utilizar también para las unidades de NAL de MVC. Sin embargo, en la MVC, la estructura de cabecera de la unidad de NAL se puede retener a excepción de las unidades de NAL prefijadas y las unidades de NAL de porciones codificadas por MVC. Las unidades de NAL de porciones codificadas por MVC pueden comprender una cabecera de cuatro bytes y la carga útil de la unidad de NAL, que puede incluir una unidad de bloque como el bloque 8 codificado de la Figura 1. Los elementos de sintaxis en la cabecera de la unidad de NAL de MVC pueden incluir `priority_id`, `temporal_id`, `anchor_pic_flag`, `view_id`, `non_idr_flag` `inter_view_flag`. En otros ejemplos, otros elementos de sintaxis se incluyen en una cabecera de la unidad de NAL de MVC.

El elemento de sintaxis `anchor_pic_flag` puede indicar si una imagen es una imagen de anclaje o imagen no anclada. Las imágenes de anclaje y todas las imágenes siguientes en el orden de salida (es decir, el orden de visualización) se pueden decodificar correctamente sin a decodificación de las imágenes anteriores en el orden de decodificación (es decir, el orden de cadena de bits) y, por lo tanto, se pueden utilizar como puntos de acceso aleatorio. Las imágenes de anclaje y las imágenes no ancladas pueden tener diferentes dependencias, que se pueden señalar en el conjunto de parámetros de secuencia.

La estructura de cadena de bits que se define en la MVC se puede caracterizar por dos elementos de sintaxis: `view_id` y `temporal_id`. El elemento de sintaxis `view_id` puede indicar el identificador de cada vista. Este identificador en la cabecera de la unidad de NAL permite una fácil identificación de las unidades de NAL en el decodificador y el acceso rápido de las vistas decodificadas para su visualización. El elemento de sintaxis `temporal_id` puede indicar la jerarquía escalabilidad temporal o, indirectamente, la velocidad de fotogramas. Por ejemplo, un punto de operación, que incluye unidades de NAL con un valor `temporal_id` máximo menor puede tener una velocidad de fotogramas más baja que el punto de operación con un valor `temporal_id` máximo mayor. Las imágenes codificadas con un valor `temporal_id` superior dependen normalmente de las imágenes codificadas con valores `temporal_id` inferiores dentro de una vista, pero pueden no depender de ninguna imagen codificada con un `temporal_id` superior.

Los elementos de sintaxis `view_id` y `temporal_id` en la cabecera de la unidad de NAL se pueden utilizar tanto para la extracción como para la adaptación de la cadena de bits. El elemento de sintaxis `priority_id` se puede utilizar principalmente por el simple procedimiento de adaptación de cadena de bits de una sola ruta. El elemento de sintaxis `inter_view_flag` puede indicar si esta unidad de NAL se utilizará para la predicción inter-vistas de otra unidad de NAL en una vista diferente.

La MVC puede también emplear conjuntos de parámetros de secuencia (SPS) e incluir una extensión MVC de SPS. Los conjuntos de parámetros se utilizan para la señalación en H.264/AVC. Los conjuntos de parámetros de secuencia comprenden información de cabecera a nivel de secuencia. Los conjuntos de parámetros de imagen (PPS) comprenden la información de cabecera a nivel de imagen con poca frecuencia cambiante. Con los conjuntos de parámetros, esta información infrecuentemente cambiante no siempre se repite para cada secuencia o imagen, por lo tanto, se mejora la eficacia de codificación. Además, el uso de conjuntos de parámetros permite la transmisión fuera de banda de la información de cabecera, evitando la necesidad de transmisiones redundantes para la capacidad de recuperación de error. En algunos ejemplos de transmisión fuera de banda, las unidades de NAL del conjunto de parámetros se transmiten en un canal diferente a las otras unidades de NAL. En la MVC, una dependencia de vista se puede señalar en la extensión MVC de SPS. Toda la predicción inter-vistas se puede hacer dentro del ámbito especificado por la extensión MVC de SPS.

En algunas técnicas de codificación de vídeos en 3D anteriores, el contenido se codifica de tal manera que los componentes de color, por ejemplo, en el espacio de color YCbCr, se codifican en una o más unidades de NAL mientras que la imagen de profundidad se codifica en una o más unidades de NAL separadas. Sin embargo, cuando no hay una unidad de NAL individual que contenga las muestras codificadas de imágenes de textura y profundidad de una unidad de acceso, pueden ocurrir diversos problemas. Por ejemplo, en un decodificador de vídeo en 3D, se espera que después de la decodificación tanto de la imagen de textura como de profundidad de cada fotograma, la representación de vista basándose en el mapa de profundidad y la textura se activa para generar las vistas virtuales. Si la unidad de NAL de la imagen profundidad y la unidad de NAL de la textura de una unidad de acceso se codifican de una manera secuencial, la representación de vista no puede comenzar hasta que toda la unidad de acceso se decodifique. Esto puede conducir a un aumento en el tiempo que toma reproducir el vídeo en 3D.

Además, la imagen de textura y la imagen del mapa de profundidad asociada pueden compartir alguna información en diversos niveles en el códec, por ejemplo, el nivel de secuencia, el nivel de imagen, nivel de porción y nivel de bloque. La codificación de esta información en dos unidades de NAL puede crear una carga extra de aplicación si comparten o predicen información. Por lo tanto, el codificador puede tener que realizar la estimación de movimiento para un fotograma dos veces, una vez para la textura y de nuevo para el mapa de profundidad. Del mismo modo, el decodificador puede necesitar realizar la compensación de movimiento dos veces para un fotograma.

Tal como se describe en la presente memoria, se añaden a las normas técnicas ya existentes, tales como MVC, con el fin de soportar vídeos en 3D. Profundidad más vídeo en múltiples vistas (MVD) se puede añadir a la MVC para el procesamiento de vídeo en 3D. Las técnicas de codificación de vídeos en 3D pueden proporcionar más flexibilidad y extensibilidad con respecto a los estándares de vídeo existentes, por ejemplo, para cambiar el ángulo de vista sin problemas o ajustar la percepción de convergencia o profundidad hacia atrás o hacia delante, lo que puede hacerse basándose en las especificaciones de los dispositivos o las preferencias del usuario, por ejemplo. Los estándares de codificación se pueden ampliar también para utilizar mapas de profundidad para la generación de vistas virtuales en vídeo en 3D.

La Figura 4 es un diagrama de flujo que ilustra una operación ejemplar de un codificador de vídeo, de acuerdo con las técnicas de la presente divulgación. En algunos ejemplos, el codificador de vídeo es un codificador de vídeo, tales como el codificador 22 de vídeo que se muestra en las Figuras 1 y 2. En otros ejemplos, el codificador de vídeo es un decodificador de vídeo, tal como el decodificador 28 de vídeo que se muestra en las Figuras 1 y 5. Un codificador de vídeo recibe una porción de textura que comprende una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción (102) de textura. Por ejemplo, un codificador de vídeo recibe una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción de textura. El procedimiento incluye además recibir una porción de profundidad que comprende una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción (104) de profundidad. Por ejemplo, el codificador de vídeo recibe una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de la información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de las características de la porción profundidad. En algunos ejemplos, el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una unidad de vista y acceso.

El procedimiento de codificación comprende además una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción (106). Por ejemplo, el codificador 22 de vídeo codifica una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la primera porción. En un ejemplo, la cabecera de porciones comprende todos los elementos de sintaxis que se utilizan para codificar la porción asociada. En otro ejemplo, decodificador 28 de vídeo descodifica una primera porción, en el que la primera porción comprende una de la porción de textura y de la porción de profundidad, en el que la primera porción tiene una cabecera de

porciones que comprende elementos de sintaxis representativos de las características de la primera porción.

El procedimiento comprende además la determinación de los elementos de sintaxis comunes para una segunda porción de la cabecera de porciones de la primera porción (108). Además, el procedimiento comprende codificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción excluyendo los valores para los elementos de sintaxis comunes a la primera porción (110). Por ejemplo, el codificador 22 de vídeo puede codificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y de la porción de profundidad diferentes de la primera porción, en el que el segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción. Del mismo modo, el decodificador 28 de vídeo puede decodificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende una de la porción de textura y de la porción de profundidad diferentes de la primera porción, en el que el segundo porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de las características de la segunda porción, excluyendo los valores de los elementos de sintaxis comunes a la primera porción.

En otros ejemplos, el procedimiento comprende además señalar una indicación de qué elementos de sintaxis están explícitamente señalados en la cabecera de porciones de la segunda porción en el conjunto de parámetros de secuencia.

En otros ejemplos, al menos un elemento de sintaxis de profundidad se determina y señala en una cabecera de porciones del componente de vista profundidad. El al menos un elemento de sintaxis de profundidad puede incluir un identificador de conjunto de parámetros de imagen, una diferencia de parámetros de cuantificación entre un parámetro de cuantificación de la porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen, la posición de partida de la unidad de bloque codificado, una orden de imágenes de referencia, o una orden de visualización de la imagen actual del componente de vista profundidad. Por ejemplo, la cabecera de porciones de la segunda porción comprende al menos un elemento de sintaxis señalado de una identificación de un conjunto de parámetros de imágenes de referencia. En otro ejemplo, la cabecera de porciones de la segunda porción comprende al menos un elemento de sintaxis señalado de una diferencia entre un parámetro de cuantificación de la segunda porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen. En otro ejemplo, la cabecera de porciones de la segunda porción comprende al menos un elemento de sintaxis señalado de una posición de partida del bloque codificado. Además, la cabecera de porciones de la segunda porción puede comprender al menos uno de un número de fotograma y un recuento de orden de imágenes de la segunda porción. En otro ejemplo, la cabecera de porciones de la segunda porción comprende al menos uno de los elementos de sintaxis relacionados con una construcción de la lista de imágenes de referencia, un número de fotogramas de referencia activos para cada lista, tablas de sintaxis de modificación de la lista de imágenes de referencia, así como tabla ponderada de predicción.

Una posición de partida de la unidad de bloque codificado se puede determinar que es cero cuando la posición de partida del bloque codificado no se indica en la cabecera de porciones de textura o en la cabecera de porciones de profundidad. Un parámetro de filtro de bucle para el al menos un componente de vista de textura se puede señalar, y un conjunto indicador que indica un parámetro de filtro de bucle utilizado para el componente de vista de profundidad es igual a un parámetro de filtro de bucle para el al menos un componente de vista de textura. Por ejemplo, la cabecera de porciones de la segunda porción comprende al menos uno de los elementos de sintaxis relacionados con parámetros de filtro de desbloqueo o parámetros de filtro de bucle adaptativos para la segunda porción.

En otro ejemplo, el uno o más bloques de datos de vídeo representativos de la información de textura se codifican utilizando predicción inter-vistas, mientras que los valores de profundidad para una parte correspondiente del fotograma se codifican utilizando predicción intra-vistas. Un fotograma de vídeo que tiene componentes de vista de textura y componentes de vista de profundidad puede corresponder a una primera vista. La codificación de uno o más bloques de datos de vídeo representativos de la información de textura puede incluir la predicción de al menos una porción de al menos uno de los bloques de datos de vídeo representativos de la información de textura con relación a los datos de una segunda vista, en el que la segunda es diferente de la primera vista. La codificación de la información de profundidad representativa de los valores de profundidad para la porción del fotograma comprende además la predicción de al menos una parte de la información de profundidad representativa de valores de profundidad en relación con los datos de la primera vista. La cabecera de porciones de profundidad puede señalar además elementos de sintaxis representativos de una construcción de la lista de imágenes de referencia para el componente de vista del mapa de profundidad.

La Tabla 1 muestra una extensión de MVC del conjunto de parámetros de secuencia (SPS). Las referencias inter-vistas se pueden señalar en el SPS y se pueden modificar por el procedimiento de construcción de la lista de imágenes de referencia, lo que permite un orden flexible de las referencias de inter-predicción o predicción o inter-vistas.

| | C | Descriptor |
|--|---|------------|
| slice_header_depth() { | | |
| if(pred_slice_header_colocated_idc < 3) | | |
| first_mb_in_slice | 2 | ue(v) |
| if(pred_slice_header_colocated_idc < 1) | | |
| slice_type | 2 | ue(v) |
| pic_parameter_set_id | 2 | ue(v) |
| if(pred_slice_header_colocated_idc < 2) | | |
| frame_num | 2 | u(v) |
| if(pred_slice_header_colocated_idc < 1) { | | |
| if(!frame_mbs_only_flag) { | | |
| field_pic_flag | 2 | u(1) |
| if(field_pic_flag) | | |
| bottom_field_flag | 2 | u(1) |
| } | | |
| if(nal_unit_type == 5) | | |
| idr_pic_id | 2 | ue(v) |
| } | | |
| if(pred_slice_header_colocated_idc < 2) { | | |
| if(pic_order_cnt_type == 0) { | | |
| pic_order_cnt_lsb | 2 | u(v) |
| if(pic_order_present_flag && !field_pic_flag) | | |
| delta_pic_order_cnt_bottom | 2 | se(v) |
| } | | |
| if(pic_order_cnt_type == 1 && !delta_pic_order_always_zero_flag) { | | |
| delta_pic_order_cnt[0] | 2 | se(v) |
| if(pic_order_present_flag && !field_pic_flag) | | |
| delta_pic_order_cnt[1] | 2 | se(v) |
| } | | |
| } | | |
| if(pred_slice_header_colocated_idc < 1) { | | |
| if(redundant_pic_cnt_present_flag) | | |
| redundant_pic_cnt | 2 | ue(v) |
| if(slice_type == B) | | |
| direct_spatial_mv_pred_flag | 2 | u(1) |
| if(slice_type == P slice_type == SP slice_type == B) { | | |
| num_ref_idx_active_override_flag | 2 | u(1) |
| if(num_ref_idx_active_override_flag) { | | |
| num_ref_idx_l0_active_minus1 | 2 | ue(v) |
| if(slice_type == B) | | |
| num_ref_idx_l1_active_minus1 | 2 | ue(v) |
| } | | |
| } | | |
| ref_pic_list_reordering() | 2 | |
| if((weighted_pred_flag && (slice_type == P slice_type == SP)) | | |
| (weighted_bipred_idc == 1 && slice_type == B)) | | |
| pred_weight_table() | 2 | |
| if(nal_ref_idc != 0) | | |
| dec_ref_pic_marking() | 2 | |
| if(entropy_coding_mode_flag && slice_type != I && slice_type != | | |
| SI) | | |
| cabac_init_idc | 2 | ue(v) |
| } | | |
| slice_qp_delta | 2 | se(v) |
| if(pred_slice_header_colocated_idc < 1) { | | |
| if(slice_type == SP slice_type == SI) { | | |
| if(slice_type == SP) | | |

(continuación)

| | | |
|---|---|-------|
| sp_for_switch_flag | 2 | u(1) |
| slice_qs_delta | 2 | se(v) |
| } | | |
| } | | |
| if(deblocking_filter_control_present_flag &&pred_slice_header_colocated_idc<1) { | | |
| disable_deblocking_filter_idc | 2 | ue(v) |
| if(disable_deblocking_filter_idc != 1) { | | |
| slice_alpha_c0_offset_div2 | 2 | se(v) |
| slice_beta_offset_div2 | 2 | se(v) |
| } | | |
| } | | |
| if(pred_slice_header_colocated_idc<1) | | |
| if(num_slice_groups_minus1 > 0 && slice_group_map_type >= 3 && slice_group_map_type <= 5) | | |
| slice_group_change_cycle | 2 | u(v) |
| } | | |
| } | | |

TABLA 1

5 Un indicador a nivel de secuencia puede especificar cómo se predicen los componentes de vista de profundidad a partir de los componentes de vista de textura correspondientes en la misma vista. En un conjunto de parámetros secuencia para un mapa de profundidad, se puede señalar la siguiente sintaxis:

pred_slice_header_colocated_idc ue(v) o u(2)

10 En los ejemplos donde el uno o más bloques de datos de vídeo representativos de la información de textura se codifican utilizando predicción inter-vistas, mientras que los valores de profundidad para una parte correspondiente del fotograma se codifican utilizando predicción intra-vistas, num_ref_idx_active_override_flag y ref_pic_list_reordering se pueden señalarse en la cabecera de porciones para los componentes de vista del mapa de profundidad.

15 La Tabla 2 proporciona una tabla de sintaxis ejemplar de una cabecera de porciones de una porción de profundidad. El elemento de sintaxis pre_slice_header_colocated_idc especifica que los elementos de sintaxis se reutilizan entre una cabecera de porciones de un componente de vista textura y una cabecera de porciones de un componente de vista profundidad de las siguientes maneras. La configuración pred_slice_header_colocated_idc igual a 0 indica que no hay predicción entre ninguna cabecera de porciones del componente de vista de textura y su correspondiente componente de vista profundidad. Tenga en cuenta que el componente de vista de textura correspondiente de un el componente de vista del mapa de profundidad se refiere al componente de vista de textura en el mismo instante de tiempo en la misma vista.

20 La configuración pre_slice_header_colocated_idc igual a 3 indica que el conjunto de parámetros de imagen y el delta QP de una unidad de NAL del componente de vista de profundidad se señalan en la cabecera de porciones, mientras que otros elementos de sintaxis al nivel de porción de la unidad de NAL del componente de vista de profundidad son iguales a o predecibles a partir de los elementos de sintaxis del componente de vista de textura correspondiente.

25 La configuración pred_slice_header_colocated_idc igual a 2 indica que el conjunto de parámetros de imagen y el delta QP, así como la ubicación de la primera MB o CU de una unidad de NAL del componente de vista de profundidad se señalan en la cabecera de porciones de profundidad, mientras que los otros elementos de sintaxis son iguales a o predecibles a partir de los elementos de sintaxis correspondientes del componente de vista de textura de ubicación conjunta de la misma vista.

30 La configuración pred_slice_header_colocated_idc igual a 1 indica que el conjunto de parámetros de imagen y el delta QP, la ubicación de la primera MB o CU de una unidad de NAL del componente de vista de profundidad y los valores frame_num y POC se señalan en la cabecera de porciones, mientras que los otros elementos de sintaxis son iguales a o predecibles a partir de los elementos de sintaxis correspondientes del componente de vista de textura de ubicación conjunta de la misma vista. En un ejemplo, cuando pred_slice_header_colocated_idc es igual a 3, se infiere que first_mb_in_slice tiene valor igual a 0. Por otra parte, cuando pred_slice_header_colocated_idc es menor que 3, un valor para first_mb_in_slice se puede señalar de manera explícita, como se muestra en la Tabla 2.

5 También se muestra en la Tabla 2, cuando `pred_slice_header_colocated_idc` tiene un valor que es menor que uno, se señala un indicador de porción entropía y un tipo de porción. El indicador de porción de entropía tiene un valor que indica si la porción correspondiente es una porción de entropía, es decir, si la porción se codifica por entropía sin referencia a los contextos de otras porciones. Los modelos de contexto pueden así inicializarse o restablecerse al inicio de cada porción de entropía. El tipo porción indica un tipo de porción, por ejemplo, I, P, o B. Además, cuando `pred_slice_header_colocated_idc` tiene un valor que es menor que uno, la cabecera de porciones indica si los bloques de la porción se codifican por campo (por ejemplo, para la codificación entrelazada por campo).

| | Descriptor |
|--|------------|
| <code>slice_header_depth() {</code> | |
| <code>if(pred_slice_header_colocated_idc < 3)</code> | |
| first_mb_in_slice | ue(v) |
| <code>if(pred_slice_header_colocated_idc < 1)</code> | |
| entropy_slice_flag | u(1) |
| <code>if(!entropy_slice_flag) {</code> | |
| <code>if(pred_slice_header_colocated_idc < 1)</code> | |
| slice_type | ue(v) |
| pic_parameter_set_id | ue(v) |
| <code>if(pred_slice_header_colocated_idc < 2)</code> | |
| frame_num | u(v) |
| <code>if(pred_slice_header_colocated_idc < 1)</code> | |
| <code>if(!IdrPicFlag)</code> | |
| idr_pic_id | ue(v) |
| <code>if(pred_slice_header_colocated_idc < 2){</code> | |
| <code>if(pic_order_cnt_type == 0)</code> | |
| pic_order_cnt_lsb /* | u(v) |
| <code>if(slice_type == P slice_type == B) {</code> | |
| num_ref_idx_active_override_flag | u(1) |
| <code>if(num_ref_idx_active_override_flag) {</code> | |
| num_ref_idx_l0_active_minus1 | ue(v) |
| <code>if(slice_type == B)</code> | |
| num_ref_idx_l1_active_minus1 | ue(v) |
| } | |
| } | |
| } | |
| <code>ref_pic_list_modification()</code> | |
| <code>ref_pic_list_combination()</code> | |
| <code>if(nal_ref_idc != 0)</code> | |
| <code>dec_ref_pic_marking()</code> | |
| <code>if(entropy_coding_mode_flag && slice_type != 1)</code> | |
| cabac_init_idc | ue(v) |
| } | |
| slice_qp_delta | se(v) |
| } | |
| <code>if(!entropy_slice_flag) {</code> | |
| <code>if(adaptive_loop_filter_enabled_flag)</code> | |
| <code>alf_param()</code> | |
| <code>if(deblocking_filter_control_present_flag</code> | |
| <code>&& pred_slice_header_colocated_idc < 1) {</code> | |
| disable_deblocking_filter_idc | |
| <code>if(disable_deblocking_filter_idc != 1) {</code> | |
| slice_alpha_c0_offset_div2 | |
| slice_beta_offset_div2 | |
| } | |
| } | |
| } | |
| ... | |
| } | |

TABLA 2

La Tabla 3 proporciona un ejemplo de diseño de una cabecera de porciones para un componente de vista la profundidad basándose en HEVC. Tenga en cuenta que en este ejemplo, cuando `pred_slice_header_colocated_idc` es igual a 3, se infiere que `first_tb_in_slice` tiene un valor igual a 0.

| | C | Descriptor |
|---|---|------------|
| <code>slice_header_depth() {</code> | | |
| <code>if (pred_slice_header_colocated_idc)</code> | | |
| <code> pred_default_syntax_flag</code> | 2 | u(1) |
| <code>if (!pred_default_syntax_flag pred_slice_header_colocated_idc <3)</code> | | |
| <code> first_mb_in_slice</code> | 2 | ue(v) |
| <code>if (!pred_default_syntax_flag)</code> | | |
| <code> slice_type</code> | 2 | ue(v) |
| <code> pic_parameter_set_id</code> | 2 | ue(v) |
| <code>if (!pred_default_syntax_flag pred_slice_header_colocated_idc <2)</code> | | |
| <code> frame_num</code> | 2 | u(v) |
| <code>if (!pred_default_syntax_flag) {</code> | | |
| <code> if (!frame_mbs_only_flag) {</code> | | |
| <code> field_pic_flag</code> | 2 | u(1) |
| <code> if (field_pic_flag) {</code> | | |
| <code> bottom_field_flag</code> | 2 | u(1) |
| <code> }</code> | | |
| <code> }</code> | | |
| <code> if (nal_unit_type == 5)</code> | | |
| <code> idr_pic_id</code> | 2 | ue(v) |
| <code> }</code> | | |
| <code>if (!pred_default_syntax_flag pred_slice_header_colocated_idc <2) {</code> | | |
| <code> if (pic_order_cnt_type == 0) {</code> | | |
| <code> pic_order_cnt_lsb</code> | 2 | u(v) |
| <code> if (pic_order_present_flag && !field_pic_flag)</code> | | |
| <code> delta_pic_order_cnt_bottom</code> | 2 | se(v) |
| <code> }</code> | | |
| <code> if (pic_order_cnt_type == 1 && !delta_pic_order_always_zero_flag) {</code> | | |
| <code> delta_pic_order_cnt[0]</code> | 2 | se(v) |
| <code> if (pic_order_present_flag && !field_pic_flag)</code> | | |
| <code> delta_pic_order_cnt[1]</code> | 2 | se(v) |
| <code> }</code> | | |
| <code>}</code> | | |
| <code>if (!pred_default_syntax_flag) {</code> | | |
| <code> if (redundant_pic_cnt_present_flag)</code> | | |
| <code> redundant_pic_cnt</code> | 2 | ue(v) |
| <code> if (slice_type == B)</code> | | |
| <code> direct_spatial_mv_pred_flag</code> | 2 | u(1) |
| <code> if (slice_type == P slice_type == SP slice_type == B) {</code> | | |
| <code> num_ref_idx_active_override_flag</code> | 2 | u(1) |
| <code> if (num_ref_idx_active_override_flag) {</code> | | |
| <code> num_ref_idx_l0_active_minus1</code> | 2 | ue(v) |
| <code> if (slice_type == B)</code> | | |
| <code> num_ref_idx_l1_active_minus1</code> | 2 | ue(v) |
| <code> }</code> | | |
| <code>}</code> | | |
| <code>ref_pic_list_reordering() </code> | 2 | |
| <code>if ((weighted_pred_flag && (slice_type == P slice_type == SP))</code> | | |
| <code> (weighted_bipred_idc == 1 && slice_type == B))</code> | | |
| <code> pred_weight_table() </code> | 2 | |
| <code>if (nal_ref_idc != 0)</code> | | |
| <code> dec_ref_pic_marking() </code> | 2 | |
| <code>if (entropy_coding_mode_flag && slice_type != I && slice_type !=</code> | | |
| <code>SI)</code> | | |
| <code> cabac_init_idc</code> | 2 | ue(v) |
| <code>}</code> | | |
| <code>slice_qp_delta</code> | 2 | se(v) |
| <code>if (!pred_default_syntax_flag) {</code> | | |

(continuación)

| | | |
|--|---|-------|
| if(slice_type == SP slice_type == SI) { | | |
| if(slice_type == SP) | | |
| sp_for_switch_flag | 2 | u(1) |
| slice_qs_delta | 2 | se(v) |
| } | | |
| } | | |
| if(deblocking_filter_control_present_flag && pred_slice_header_colocated_idc < 1) { | | |
| disable_deblocking_filter_idc | 2 | ue(v) |
| if(disable_deblocking_filter_idc != 1) { | | |
| slice_alpha_c0_offset_div2 | 2 | se(v) |
| slice_beta_offset_div2 | 2 | se(v) |
| } | | |
| } | | |
| if(!pred_default_syntax_flag) | | |
| if(num_slice_groups_minus1 > 0 && slice_group_map_type >= 3 && slice_group_map_type <= 5) | | |
| slice_group_change_cycle | 2 | u(v) |
| } | | |
| } | | |
| } | | |

TABLA 3

5 La Tabla 4 es una tabla de sintaxis ejemplar de una cabecera de porciones de una porción de profundidad. La Tabla 4 proporciona un ejemplo de diseño de una sintaxis de cabecera de porciones profundidad para indicar además la reutilización de sintaxis para un componente de vista profundidad. En este ejemplo, un indicador al nivel de secuencia específica cómo se predicen los componentes de vista la profundidad a partir de los componentes de vista de textura correspondientes en la misma vista.

10 En un conjunto de parámetros de secuencia de este tipo para el mapa de profundidad, se puede señalar la siguiente sintaxis:

pred_slice_header_colocated_idc ue(v) o u(2)

15 El elemento de sintaxis pre_slice_header_colocated_idc especifica la reutilización del elemento de sintaxis entre la cabecera de porciones de un componente de vista de textura y la cabecera de porciones de un componente de vista profundidad. Por ejemplo, establecer pred_slice_header_colocated_idc igual a 0 indica que no hay predicción entre ninguna cabecera de porciones del componente de vista de textura y su correspondiente componente de vista profundidad. La configuración pre_slice_header_colocated_idc igual a 3 indica que el conjunto de parámetros de imagen y el delta QP de una unidad de NAL del componente de vista de profundidad se señalan en la cabecera de porciones, mientras que los otros elementos de sintaxis al nivel de porción de la unidad de NAL del componente de vista de profundidad son iguales a 0 predecibles a partir de los elementos de sintaxis del componente de vista de textura correspondiente.

20 La configuración pred_slice_header_colocated_idc igual a 2 indica que el conjunto de parámetros de imagen y el delta QP, así como la ubicación de la primera MB o CU de una unidad de NAL del componente de vista de profundidad se señalan en la cabecera de porciones, mientras que los otros elementos de sintaxis son iguales a 0 predecibles a partir de los elementos de sintaxis correspondientes del componente de vista de textura de ubicación conjunta de la misma vista. La configuración pred_slice_header_colocated_idc igual a 1 indica que el conjunto de parámetros de imagen y el delta QP, la ubicación de la primera MB o CU de una unidad de NAL del componente de vista de profundidad y los valores frame_num y POC se señalan en la cabecera de porciones, mientras que los otros elementos de sintaxis son iguales a 0 predecibles a partir de los elementos de sintaxis correspondientes del componente de vista de textura de ubicación conjunta de la misma vista.

25 Un indicador de sintaxis, pred_default_syntax_flag, indica si los elementos de sintaxis de la cabecera de porciones de un componente de vista de mapa de profundidad se predicen a partir de aquellos del componente de vista de textura de ubicación conjunta. En un ejemplo, se infiere que pred_default_syntax_flag es 0, si pred_slice_header_colocated_idc es igual a 0. Cuando pred_slice_header_colocated_idc es igual a 3 y pred_default_syntax_flag es 1 en este ejemplo, firs_mb_in_slice es igual a 0.

35

| | C | Descriptor |
|---|---|------------|
| slice_header_depth() { | | |
| if (pred_slice_header_colocated_idc) | | |
| pred_default_syntax_flag | 2 | u(1) |
| if (!pred_default_syntax_flag pred_slice_header_colocated_idc <3) | | |
| first_mb_in_slice | 2 | ue(v) |
| if (!pred_default_syntax_flag) | | |
| slice_type | 2 | ue(v) |
| pic_parameter_set_id | 2 | ue(v) |
| if (!pred_default_syntax_flag pred_slice_header_colocated_idc <2) | | |
| frame_num | 2 | u(v) |
| if (!pred_default_syntax_flag) { | | |
| if (!frame_mbs_only_flag) { | | |
| field_pic_flag | 2 | u(1) |
| if (field_pic_flag) | | |
| bottom_field_flag | 2 | u(1) |
| } | | |
| if (nal_unit_type == 5) | | |
| idr_pic_id | 2 | ue(v) |
| } | | |
| if (!pred_default_syntax_flag pred_slice_header_colocated_idc <2) { | | |
| if (pic_order_cnt_type == 0) { | | |
| pic_order_cnt_lsb | 2 | u(v) |
| if (pic_order_present_flag && !field_pic_flag) | | |
| delta_pic_order_cnt_bottom | 2 | se(v) |
| } | | |
| if (pic_order_cnt_type == 1 && !delta_pic_order_always_zero_flag) { | | |
| delta_pic_order_cnt[0] | 2 | se(v) |
| if (pic_order_present_flag && !field_pic_flag) | | |
| delta_pic_order_cnt[1] | 2 | se(v) |
| } | | |
| } | | |
| if (!pred_default_syntax_flag) { | | |
| if (redundant_pic_cnt_present_flag) | | |
| redundant_pic_cnt | 2 | ue(v) |
| if (slice_type == B) | | |
| direct_spatial_mv_pred_flag | 2 | u(1) |
| if (slice_type == P slice_type == SP slice_type == B) { | | |
| num_ref_idx_active_override_flag | 2 | u(1) |
| if (num_ref_idx_active_override_flag) { | | |
| num_ref_idx_l0_active_minus1 | 2 | ue(v) |
| if (slice_type == B) | | |
| num_ref_idx_l1_active_minus1 | 2 | ue(v) |
| } | | |
| } | | |
| } | | |
| ref_pic_list_reordering() | 2 | |
| if ((weighted_pred_flag && (slice_type == P slice_type == SP)) | | |
| (weighted_bipred_idc == 1 && slice_type == B)) | | |
| pred_weight_table() | 2 | |
| if (nal_ref_idc != 0) | | |
| dec_ref_pic_marking() | 2 | |
| if (entropy_coding_mode_flag && slice_type != I && slice_type != SI) | | |
| cabac_init_idc | 2 | ue(v) |
| } | | |
| slice_qp_delta | 2 | se(v) |
| if (!pred_default_syntax_flag) { | | |

(continuación)

| | | |
|--|---|-------|
| if(slice_type == SP slice_type == SI) { | | |
| if(slice_type == SP) | | |
| sp_for_switch_flag | 2 | u(1) |
| slice_qs_delta | 2 | se(v) |
| } | | |
| } | | |
| if(deblocking_filter_control_present_flag && pred_slice_header_colocated_idc < 1) { | | |
| disable_deblocking_filter_idc | 2 | ue(v) |
| if(disable_deblocking_filter_idc != 1) { | | |
| slice_alpha_c0_offset_div2 | 2 | se(v) |
| slice_beta_offset_div2 | 2 | se(v) |
| } | | |
| } | | |
| if(!pred_default_syntax_flag) | | |
| if(num_slice_groups_minus1 > 0 && slice_group_map_type >= 3 && slice_group_map_type <= 5) | | |
| slice_group_change_cycle | 2 | u(v) |
| } | | |
| } | | |

TABLA 4

5 La Tabla 5 es una tabla de sintaxis ejemplar de una cabecera de porciones para un componente de vista de profundidad basándose en HEVC. En el ejemplo de la Tabla 5, pred_default_syntax_flag indica si los elementos de sintaxis de la cabecera de porciones de un componente de vista del mapa de profundidad se predicen a partir del componente de vista de textura de ubicación conjunta. Se infiere que el indicador pred_default_syntax_flag es 0 si pred_slice_header_colocated_idc es igual a 0. Cuando pred_slice_header_colocated_idc es igual a 3 y pred_default_syntax_flag es 1, en este ejemplo, first_tb_in_slice es igual a 0.

10

| | Descriptor |
|---|------------|
| slice_header_depth() { | |
| if(pred_slice_header_colocated_idc) | |
| pred_default_syntax_flag | u(1) |
| if(pred_slice_header_colocated_idc < 3 !pred_default_syntax_flag) | |
| first_tb_in_slice | ue(v) |
| if(!pred_default_syntax_flag) | |
| entropy_slice_flag | u(1) |
| if(!entropy_slice_flag) { | |
| if(!pred_default_syntax_flag) | |
| slice_type | ue(v) |
| pic_parameter_set_id | ue(v) |
| if(!pred_default_syntax_flag pred_slice_header_colocated_idc < 2) | |
| frame_num | u(v) |
| if(!pred_default_syntax_flag) | |
| if(IdrPicFlag) | |
| idr_pic_id | ue(v) |
| if(!pred_default_syntax_flag) { | |
| if(pic_order_cnt_type == 0) | |
| pic_order_cnt_lsb /* | u(v) |
| if(slice_type == P slice_type == B) { | |
| num_ref_idx_active_override_flag | u(1) |
| if(num_ref_idx_active_override_flag) { | |
| num_ref_idx_l0_active_minus1 | ue(v) |
| if(slice_type == B) | |
| num_ref_idx_l1_active_minus1 | ue(v) |
| } | |
| } | |
| ref_pic_list_modification() | |
| ref_pic_list_combination() | |
| if(nal_ref_idc != 0) | |
| dec_ref_pic_marking() | |
| if(entropy_coding_mode_flag && slice_type != I) | |
| cabac_init_idc | ue(v) |
| } | |
| slice_qp_delta | se(v) |
| } | |
| if(!entropy_slice_flag) { | |
| if(adaptive_loop_filter_enabled_flag) | |
| alf_param() | |
| if(deblocking_filter_control_present_flag !pred_default_syntax_flag) { | |
| disable_deblocking_filter_idc | |
| if(disable_deblocking_filter_idc != 1) { | |
| slice_alpha_c0_offset_div2 | |
| slice_beta_offset_div2 | |
| } | |
| } | |
| } | |
| ... | |
| } | |

TABLA 5

Tenga en cuenta que cuando se habilita la predicción de la cabecera de porciones, hay una implicación de que si la porción A se basa en la porción B, siempre que, ya sea, la porción A o B sea una porción de profundidad y la otra una porción de textura y que pertenezcan a la vista del mismo instante de tiempo, uno de las siguientes se cumple:

5

todas las porciones en la imagen que contienen la porción B tienen la misma cabecera de porciones; cualquier MB en la porción A tiene un MB de ubicación conjunta en la porción B; o si algún MB en la porción A tiene un MB de ubicación conjunta en la porción C de la imagen que contiene la porción B, la porción C debe tener la misma cabecera de porciones que la porción B.

- 5 Como alternativa, una implementación diferente de la técnica descrita puede ser como sigue para un componente de vista profundidad. La Tabla 6 proporciona un ejemplo de una extensión de profundidad de la cabecera de porciones.

| | C | Descriptor |
|--|----------|-------------------|
| slice_header_depth_extension() { | | |
| if(pred_slice_header_depth_idc == 0) | | |
| slice_header() | | |
| else { | | |
| if(pred_slice_header_depth_idc == 2) | | |
| first_mb_in_slice | 2 | ue(v) |
| pic_parameter_set_id | 2 | ue(v) |
| if (sameRefPicList) { | | |
| if(slice_type == P slice_type == SP slice_type == B) { | | |
| num_ref_idx_active_override_flag | 2 | u(1) |
| if(num_ref_idx_active_override_flag) { | | |
| num_ref_idx_l0_active_minus1 | 2 | ue(v) |
| if(slice_type == B) | | |
| num_ref_idx_l1_active_minus1 | 2 | ue(v) |
| } | | |
| } | | |
| ref_pic_list_mvc_modification() | 2 | |
| } | | |
| slice_qp_delta | 2 | se(v) |
| } | | |
| } | | |
| } | | |

TABLA 6

- 10 En este ejemplo, el elemento de sintaxis sameRefPicList se deriva de o señala al nivel de SPS o PPS. Por ejemplo, un disable_depth_inter_view_flag, señalado en el SPS, indica si la predicción inter-vistas para la profundidad está deshabilitada.

- 15 Para un componente de vista de textura, otra implementación de la técnica descrita puede ser como se muestra en la Tabla 7. En este ejemplo, los elementos de sintaxis para una cabecera de porciones de textura para los componentes de vista de textura se pueden predecir a partir de elementos de sintaxis correlacionados para una cabecera de porciones de profundidad para los componentes de vista de profundidad.

| | C | Descriptor |
|--|----------|-------------------|
| slice_header_texture_extension() { | | |
| if(pred_slice_header_depth_idc == 0) | | |
| slice_header() | | |
| else { | | |
| if(pred_slice_header_depth_idc == 2) | | |
| first_mb_in_slice | 2 | ue(v) |
| pic_parameter_set_id | 2 | ue(v) |
| if (!sameRefPicList) { | | |
| if(slice_type == P slice_type == SP slice_type == B) { | | |

(continuación)

| | | |
|--|---|-------|
| num_ref_idx_active_override_flag | 2 | u(1) |
| if(num_ref_idx_active_override_flag) { | | |
| num_ref_idx_l0_active_minus1 | 2 | ue(v) |
| if(slice_type == B) | | |
| num_ref_idx_l1_active_minus1 | 2 | ue(v) |
| } | | |
| } | | |
| ref_pic_list_mvc_modification() | 2 | |
| } | | |
| slice_qp_delta | 2 | se(v) |
| } | | |
| } | | |

TABLA 7

Asimismo, en este ejemplo, el elemento de sintaxis sameRefPicList se deriva de o señala en un nivel de SPS o PPS.

- 5 Como alternativa, un indicador de este tipo se puede señalar explícitamente en la cabecera de porciones como se muestra en la Tabla 8.

| | C | Descriptor |
|--|----------|-------------------|
| slice_header_texture_extension() { | | |
| if(pred slice_header_depth_idc == 0) | | |
| slice_header() | | |
| else { | | |
| slice_header_prediction_flag | 2 | u(1) |
| if(pred slice_header_depth_idc == 2) | | |
| first_mb_in_slice | 2 | ue(v) |
| pic_parameter_set_id | 2 | ue(v) |
| if(!sameRefPicList) { | | |
| if(slice_type == P slice_type == SP slice_type == B) { | | |
| num_ref_idx_active_override_flag | 2 | u(1) |
| if(num_ref_idx_active_override_flag) { | | |
| num_ref_idx_l0_active_minus1 | 2 | ue(v) |
| if(slice_type == B) | | |
| num_ref_idx_l1_active_minus1 | 2 | ue(v) |
| } | | |
| } | | |
| ref_pic_list_mvc_modification() | 2 | |
| } | | |
| slice_qp_delta | 2 | se(v) |
| } | | |
| } | | |

TABLA 8

- 10 Un elemento de sintaxis, slice_header_prediction_flag, indica si la predicción la cabecera de porciones de textura a profundidad o de profundidad a textura está habilitada. Es decir, al menos una de la porción de textura o la porción de profundidad comprende un elemento de sintaxis que indica si una predicción de la cabecera de porciones es de la cabecera de porciones de textura a la cabecera de porciones de profundidad o de la cabecera de porciones de profundidad a la cabecera de porciones de textura.

- 15 Como alternativa, indicadores del nivel de porción u otros indicadores especifican en qué medida se aplica la predicción de porción. Ejemplos de estos indicadores incluyen si se predicen los elementos de sintaxis de la construcción de la lista de imágenes de referencia, si se predice slice_qp_delta, y si se predicen los elementos de sintaxis de predicción ponderados.

En algunos ejemplos, se ha indicado también si se predicen elementos de sintaxis relacionados con el filtro de bucle. Si no se predicen elementos de sintaxis relacionados con el filtro de bucle, un indicador adicional para indicar si los

elementos de sintaxis están presentes o no se incluye en la cabecera de porciones de profundidad.

Como alternativa, otro indicador utilizado para señalar un filtro de desbloqueo, `deblocking_pred_flag`, puede utilizarse en lugar de `pred_default_syntax_flag` o `pred_slice_header_colocated_idc` para los parámetros de filtro de desbloqueo. Este indicador se señala en la misma cabecera de porciones o PPS o SPS. La Tabla 9 muestra una tabla de sintaxis ejemplar de una cabecera de porciones para un componente de vista la profundidad basándose en HEVC. En el contexto de HEVC, no se supone que los parámetros de ALF de un componente de vista profundidad sean los mismos que los parámetros de ALF del componente de vista de textura correspondiente, a menos que ALF no se utilice ni para el componente de vista textura ni para el componente de vista de profundidad.

5

| slice header depth() { | C | Descriptor |
|---|---|------------|
| if(pred slice header colocated_idc) | | |
| pred_default_syntax_flag | 2 | u(1) |
| ... | | |
| if(deblocking_filter_control_present_flag && !deblocking_pred_flag) { | | |
| disable_deblocking_filter_idc | 2 | ue(v) |
| if(disable_deblocking_filter_idc != 1) { | | |
| slice_alpha_c0_offset_div2 | 2 | se(v) |
| slice_beta_offset_div2 | 2 | se(v) |
| } | | |
| } | | |
| ... | | |
| } | | |

10

TABLA 9

La Figura 5 es un diagrama de bloques que ilustra un ejemplo del decodificador 28 de vídeo de la Figura 1 con más detalle, de acuerdo con las técnicas de la presente divulgación. El decodificador 28 de vídeo es un ejemplo de un dispositivo o aparato informático de vídeo especializado denominado en la presente memoria como un "codificador". Como se muestra en la Figura 5, el decodificador 28 de vídeo corresponde al decodificador 28 de vídeo del dispositivo 14 de destino. Sin embargo, en otros ejemplos, el decodificador 28 de vídeo corresponde a un dispositivo diferente. En otros ejemplos, otras unidades (tales como, por ejemplo, otro codificador/decodificador (CODECS)) pueden realizar también técnicas similares al decodificador 28 de vídeo.

15

El decodificador 28 de vídeo incluye una unidad 52 de codificación por entropía que decodifica por entropía la cadena de bits recibidos para generar coeficientes cuantificados y los elementos de sintaxis de predicción. La cadena de bits incluye bloques que tienen componentes de textura y un componente de profundidad para cada posición de pixel codificada a fin de proporcionar un vídeo en 3D y elementos de sintaxis. Los elementos de sintaxis de predicción incluyen al menos uno de un modo de codificación, uno o más vectores de movimiento, información que identifica la técnica de interpolación utilizada, coeficientes para su uso en el filtrado de interpolación, y otra información asociada con la generación del bloque de predicción.

20

Los elementos de sintaxis de predicción, por ejemplo, los coeficientes, se reenvían a la unidad 55 de procesamiento de predicción. La unidad 55 de procesamiento de predicción incluye un módulo de predicción de sintaxis de profundidad 66. Si la predicción se utiliza para codificar los coeficientes relativos a los coeficientes de un filtro fijo, o uno en relación con otro, la unidad 55 de procesamiento de predicción decodifica los elementos de sintaxis para definir los coeficientes reales. El módulo 66 de predicción de sintaxis de profundidad predice elementos de sintaxis de profundidad para los componentes de vista profundidad a partir de los elementos de sintaxis de textura de los componentes de vista de textura.

25

30

Si la cuantificación se aplica a cualquiera de los elementos de sintaxis de predicción, la unidad 56 de cuantificación inversa elimina tal cuantificación. La unidad 56 de cuantificación inversa puede tratar los componentes de profundidad y textura para cada posición de pixel de los bloques codificados en la cadena de bits codificados de manera diferente. Por ejemplo, cuando el componente de profundidad se cuantifica de manera diferente que los componentes de textura, la unidad 56 de cuantificación inversa procesa los componentes de profundidad y de textura por separado. Coeficientes de filtro, por ejemplo, se pueden codificar de manera predictiva y cuantificarse de acuerdo con la presente divulgación, y en este caso, la unidad 56 de cuantificación inversa se utiliza por el decodificador 28 de vídeo para cuantificar y de-cuantificar de forma predecible tales coeficientes.

35

La unidad 55 de procesamiento de predicción genera datos de predicción basándose en los elementos de sintaxis de predicción y uno o más bloques previamente decodificados que se almacenan en la memoria 62, en la misma manera en que se ha descrito en detalle anteriormente con respecto a la unidad 32 de procedimiento de predicción del codificador 22 de vídeo. En particular, la unidad 55 de procesamiento de predicción realiza una o más de las técnicas de vídeo más profundidad en múltiples vistas de la presente divulgación durante la compensación de movimiento para generar un bloque de predicción que incorpora componentes de profundidad, así como componentes de textura. El bloque de predicción (así como un bloque codificado) puede tener diferente resolución

40

45

para los componentes de profundidad en comparación con los componentes de textura. Por ejemplo, los componentes de profundidad tienen una precisión de un cuarto de píxel, mientras que los componentes de textura tienen una precisión de un píxel entero. Como tal, una o más de las técnicas de la presente divulgación se utilizan por el decodificador 28 de vídeo en la generación de un bloque de predicción. En algunos ejemplos, la unidad 55 de procesamiento de predicción puede incluir una unidad de compensación de movimiento que comprende filtros utilizados para la interpolación y técnicas de filtrado similares a la interpolación de la presente divulgación. El componente de compensación de movimiento no se muestra en la Figura 5 por simplicidad y facilidad de ilustración.

La unidad 56 de cuantificación inversa cuantifica de forma inversa, es decir, de-cuantifica, los coeficientes cuantificados. El procedimiento de cuantificación inversa es un procedimiento definido para la decodificación H.264 o para cualquier otro estándar de decodificación. La unidad 58 de procesamiento de transformación inversa aplica una transformación inversa, por ejemplo, una DCT inversa o procedimiento de transformación inversa conceptualmente similar, a los coeficientes de transformación con el fin de producir bloques residuales en el dominio de píxeles. El sumador 64 suma el bloque residual con el bloque de predicción correspondiente generado por la unidad 55 de procesamiento de predicción para formar una versión reconstruida del bloque original codificado por el codificador 22 de vídeo. Si se desea, también se aplica un filtro de desbloqueo para filtrar los bloques decodificados a fin de eliminar artefactos de autoarranque. Los bloques de vídeo decodificados se almacenan en la memoria 62, proporcionando bloques de referencia para la compensación de movimiento posterior y produce también vídeo decodificado para accionar el dispositivo de visualización (tal como el dispositivo 28 de la Figura 1).

El vídeo decodificado se puede utilizar para proporcionar un vídeo en 3D. El vídeo en 3D puede comprender una vista virtual tridimensional. La información de profundidad se utiliza para determinar un desplazamiento horizontal (disparidad horizontal) para cada píxel en el bloque. La manipulación de oclusión se puede realizar también para generar la vista virtual. Elementos de sintaxis para los componentes de vista de profundidad se pueden predecir a partir de los elementos de sintaxis para los componentes de vista de textura.

La Figura 6 es un diagrama de flujo que ilustra una operación ejemplar de un decodificador de vídeo, de acuerdo con las técnicas de la presente divulgación. El procedimiento de la Figura 6 se puede considerar como el procedimiento de decodificación recíproco al procedimiento de codificación de la Figura 4. La Figura 6 se describirá desde la perspectiva del decodificador 28 de vídeo de la Figura 5, aunque otros dispositivos pueden realizar técnicas similares.

Un decodificador de vídeo, tal como el decodificador 28 de vídeo, recibe una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de la información de textura de al menos una porción de un fotograma de los datos de vídeo, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de las características de la porción (122) de textura. El decodificador de vídeo recibe una porción de profundidad para un componente de vista de profundidad correspondiente al componente de vista de textura, comprendiendo la porción de profundidad la información de profundidad codificada y una cabecera de porciones de profundidad que comprende al menos un elemento de sintaxis representativo de las características de la porción de profundidad, excluyendo los valores de los elementos de sintaxis comunes en la porción de profundidad y en la porción (124) de textura. El decodificador de vídeo predice elementos de sintaxis para al menos una de la porción de profundidad o la porción de textura a partir de los valores de los elementos de sintaxis comunes en la porción de profundidad y en porción (126) de textura.

En otros ejemplos, al menos un elemento de sintaxis de profundidad se determina y señala en una cabecera de porciones del componente de vista profundidad. El al menos un elemento de sintaxis de profundidad incluye al menos uno de un identificador del conjunto de parámetros de imagen, una diferencia del parámetro de cuantificación entre un parámetro de cuantificación de la porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen, una posición de partida de la unidad de bloque codificado, un orden de las imágenes de referencia y un orden de visualización de la imagen actual del componente de vista profundidad. Se determina que la posición de partida de la unidad de bloque codificado es cero cuando la posición de partida del bloque codificado no se señala en la cabecera de porciones de textura o en la cabecera de porciones de profundidad. Un parámetro de filtro de bucle para el al menos un componente de vista de textura se puede señalar, y un conjunto de indicadores que indica un parámetro de filtro de bucle utilizado para el componente de vista de profundidad es el mismo que un parámetro de filtro de bucle para el al menos un componente de vista de textura.

En otro ejemplo, el decodificador 28 de vídeo predice el componente de vista de textura utilizando técnicas de predicción inter-vistas y predice el componente de vista profundidad utilizando técnicas de predicción intra-vistas. El decodificador 28 de vídeo recibe la cabecera de porciones de profundidad que comprende además elementos de sintaxis representativos de una construcción de la lista de imágenes de referencia para el componente de vista profundidad. En un ejemplo donde el componente de vista de textura y el componente de vista de profundidad corresponden a una primera vista, la decodificación de la componente de vista de textura incluye la predicción de al menos una porción del componente de vista textura relativa a los datos de una segunda vista. La segunda vista es diferente de la primera vista. En algunos ejemplos, la decodificación del componente de vista de profundidad puede incluir la predicción de al menos una porción de la componente de vista profundidad relativa a los datos de la primera vista.

En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware, o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse en o transmitirse a través de como una o más instrucciones o código en un medio legible por ordenador y ejecutarse por una unidad de procesamiento basándose en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, lo que corresponde a un medio tangible tal como un medio de almacenamiento de datos, o medios de comunicación incluyendo 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 generalmente pueden corresponder a (1) medios de almacenamiento tangibles legibles por ordenador que no son transitorios o (2) un medio de comunicación tal como una onda portadora o señal. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder por uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la aplicación de las técnicas descritas en la presente divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

A modo de ejemplo, y no de limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, memoria flash, o cualquier otro medio que se pueda utilizar para almacenar el código del programa deseado en forma de instrucciones o estructuras de datos y que se pueda acceder por un ordenador. También, cualquier conexión se denomina correctamente como medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten de una página web, el servidor u otra fuente remota mediante un cable coaxial, cable de fibra óptica, par trenzado, línea de suscriptor digital (DSL), o tecnologías inalámbricas como infrarrojos, radio y microondas, entonces el cable coaxial, cable de fibra óptica, par trenzado, DSL o tecnologías inalámbricas como infrarrojos, radio y microondas se incluyen en la definición del medio. Sin embargo, se debe entender, 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 se dirigen a medios de almacenamiento tangibles, no transitorios. El disco o diskette, como se utiliza en la presente memoria, incluye disco compacto (CD), discos láser, discos ópticos, discos versátiles digitales (DVD), diskettes y disco de Blu-ray donde los diskettes reproducen generalmente datos de forma magnética, mientras que los discos reproducen datos ópticamente con láser. Combinaciones de los anteriores deben incluirse también dentro del alcance de medios legibles por ordenador.

Las instrucciones se pueden ejecutar por uno o más procesadores, tales como uno o más procesadores digitales de señal (DSP), microprocesadores de propósito general, circuitos integrados de aplicación específica (ASIC), matrices campo de lógica programable (FPGA), u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, el término "procesador", como se utiliza en la presente memoria se puede referir a cualquiera de la estructura anterior o cualquier otra estructura adecuada para la aplicación de las técnicas descritas en la presente memoria. Además, en algunos aspectos, la funcionalidad descrita en la presente memoria se puede proporcionar dentro de los módulos de hardware y/o software, dedicados y configurados para codificar y decodificar, o incorporarse en un codec combinado. Además, las técnicas podrían aplicarse plenamente en uno o más circuitos o elementos lógicos.

Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de circuitos integrados (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en la presente divulgación para enfatizar los aspectos funcionales de los dispositivos configurados para realizar las técnicas descritas, pero no necesariamente requieren la realización de diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware codec o proporcionarse por una colección de unidades de hardware interoperativas, incluyendo uno o más procesadores como se ha descrito anteriormente, junto con el software y/o firmware adecuados.

Se han descrito diversos ejemplos de la presente divulgación. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

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

5 recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de características de la porción de textura;

10 recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una vista y a una unidad de acceso;

15 decodificar una primera porción, en el que la primera porción comprende la porción de textura, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la primera porción;

determinar elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción; y

20 decodificar la segunda porción después de la decodificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende la porción de profundidad, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la segunda porción, excluyendo valores para los elementos de sintaxis que son comunes con la primera porción.

2. El procedimiento de la reivindicación 1, en el que la cabecera de porciones de la segunda porción comprende uno o más de:

25 un elemento de sintaxis señalado de una identificación de un conjunto de parámetros de una imagen de referencia;

un elemento de sintaxis señalado de una diferencia de parámetros de cuantificación entre un parámetro de cuantificación de la segunda porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen;

30 un elemento de sintaxis señalado de una posición de partida de uno de los bloques codificados;

un número de fotograma;

un recuento del orden de imágenes de la segunda porción;

elementos de sintaxis relacionados con una construcción de la lista de imágenes de referencia;

un número de fotogramas de referencia activos para cada lista;

35 una tabla de sintaxis de modificación de la lista de imágenes de referencia;

una tabla de ponderación de predicción; y

elementos de sintaxis relacionados con parámetros de filtro de desbloqueo o parámetros de filtrado de bucle adaptativo para la segunda porción.

3. El procedimiento de cualquiera de las reivindicaciones 1-2, que comprende además:

40 determinar que una posición de partida de la porción profundidad es cero cuando una posición de partida del componente de vista de profundidad no se señala en la cabecera de porciones de textura o en la cabecera de porciones de profundidad.

4. Un dispositivo para la decodificación de datos de vídeo, que comprende:

45 medios para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques codificados de datos de vídeo representativos de información de textura, comprendiendo la porción de textura el uno o más bloques codificados y una cabecera de porciones textura que comprende elementos de sintaxis representativos de características de la porción de textura;

50 medios para recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques codificados de información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques codificados de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una vista y a una unidad acceso;

55 medios para decodificar una primera porción, en el que la primera porción comprende la porción de textura, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la primera porción;

medios para determinar los elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción; y

medios para decodificar la segunda porción después de la decodificación de la primera porción, al menos

parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende la porción de profundidad, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la segunda porción, excluyendo valores para los elementos de sintaxis que son comunes a la primera porción.

5 5. El dispositivo de la reivindicación 4, en el que la cabecera de porciones de la segunda porción comprende uno o más de:

un elemento de sintaxis señalado de una identificación de un conjunto de parámetros de una imagen de referencia;

10 un elemento de sintaxis señalado de una diferencia de parámetros de cuantificación entre un parámetro de cuantificación de la segunda porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen;

un elemento de sintaxis señalado de una posición de partida de uno de los bloques codificados;

un número de fotograma;

15 un recuento del orden de imágenes de la segunda porción;

elementos de sintaxis relacionados con una construcción de la lista de imágenes de referencia;

un número de fotogramas de referencia activos para cada lista;

una tabla de sintaxis de modificación de la lista de imágenes de referencia;

una tabla de ponderación de predicción; y

20 elementos de sintaxis relacionados con parámetros de filtro de desbloqueo o parámetros de filtrado de bucle adaptativo para la segunda porción.

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

recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques de datos de vídeo representativos de información de textura, comprendiendo la porción de textura el uno o más bloques y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de características de la porción de textura;

25 recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques de información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una vista y a una unidad de acceso;

codificar una primera porción, en el que la primera porción comprende la porción de textura,

en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la primera porción;

35 determinar los elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción; y

codificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende la porción de profundidad, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la segunda porción, excluyendo valores para los elementos de sintaxis que son comunes a la primera porción.

7. El procedimiento de la reivindicación 6, en el que la cabecera de porciones de la segunda porción comprende uno o más de:

45 un elemento de sintaxis señalado de una identificación de un conjunto de parámetros de una imagen de referencia;

un elemento de sintaxis señalado de una diferencia de parámetros de cuantificación entre un parámetro de cuantificación de la segunda porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen;

un elemento de sintaxis señalado de una posición de partida de uno de los bloques codificados;

50 un número de fotograma;

un recuento del orden de imágenes de la segunda porción;

elementos de sintaxis relacionados con una construcción de la lista de imágenes de referencia;

un número de fotogramas de referencia activos para cada lista;

una tabla de sintaxis de modificación de la lista de imágenes de referencia;

55 una tabla de ponderación de predicción; y

elementos de sintaxis relacionados con parámetros de filtro de desbloqueo o parámetros de filtrado de bucle adaptativo para la segunda porción.

8. El procedimiento de cualquiera de las reivindicaciones 6-7, que comprende además:

determinar que una posición de partida de la porción profundidad es cero cuando una posición de partida del componente de vista de profundidad no se señala en la cabecera de porciones de textura o en la cabecera de porciones de profundidad.

5 9. El procedimiento de cualquiera de las reivindicaciones 6-8, que comprende además:

señalar una indicación de qué elementos de sintaxis están explícitamente señalados en la cabecera de porciones de la segunda porción en el conjunto de parámetros de secuencia.

10. Un dispositivo para la codificación de datos de vídeo, que comprende:

10 medios para recibir una porción de textura para un componente de vista de textura asociado con uno o más bloques de datos de vídeo representativos de información de textura, comprendiendo la porción de textura el uno o más bloques y una cabecera de porciones de textura que comprende elementos de sintaxis representativos de características de la porción de textura;

15 medios para recibir una porción de profundidad para un componente de vista de profundidad asociado con uno o más bloques de información de profundidad correspondiente al componente de vista de textura, en el que la porción de profundidad comprende el uno o más bloques de información de profundidad y una cabecera de porciones de profundidad que comprende elementos de sintaxis representativos de características de la porción de profundidad, y en el que el componente de vista de profundidad y el componente de vista de textura pertenecen ambos a una vista y a una unidad de acceso;

20 medios para codificar una primera porción, en el que la primera porción comprende la porción de textura, en el que la primera porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la primera porción;

medios para determinar los elementos de sintaxis comunes para una segunda porción a partir de la cabecera de porciones de la primera porción; y

25 medios para codificar la segunda porción después de la codificación de la primera porción, al menos parcialmente basándose en los elementos de sintaxis comunes determinados, en el que la segunda porción comprende la porción de profundidad, en el que la segunda porción tiene una cabecera de porciones que comprende elementos de sintaxis representativos de características de la segunda porción, sin repetir valores para los elementos de sintaxis que son comunes a la primera porción.

30 11. El dispositivo de la reivindicación 10, en el que la cabecera de porciones de la segunda porción comprende uno o más de:

un elemento de sintaxis señalado de una identificación de un conjunto de parámetros de una imagen de referencia;

35 un elemento de sintaxis señalado de una diferencia de parámetros de cuantificación entre un parámetro de cuantificación de la segunda porción y un parámetro de cuantificación señalado en un conjunto de parámetros de imagen;

un elemento de sintaxis señalado de una posición de partida de uno de los bloques codificados;

un número de fotograma;

un recuento del orden de imágenes de la segunda porción;

40 elementos de sintaxis relacionados con una construcción de la lista de imágenes de referencia;

un número de fotogramas de referencia activos para cada lista;

una tabla de sintaxis de modificación de la lista de imágenes de referencia;

una tabla de ponderación de predicción; y

elementos de sintaxis relacionados con parámetros de filtro de desbloqueo o parámetros de filtrado de bucle adaptativo para la segunda porción.

45 12. El dispositivo de cualquiera de las reivindicaciones 10-11, que comprende además:

medios para señalar una indicación de qué elementos de sintaxis están explícitamente señalados en la cabecera de porciones de la segunda porción en el conjunto de parámetros de secuencia.

50 13. Un medio de almacenamiento legible por ordenador que tiene almacenado en su interior instrucciones que, cuando son ejecutadas, hacen que un procesador de un dispositivo de descodificación de vídeos realice el procedimiento de cualquier combinación de las reivindicaciones 1-3 o hacen que un procesador de un dispositivo de descodificación de vídeos realice el procedimiento de cualquier combinación de las reivindicaciones 6-9.

10

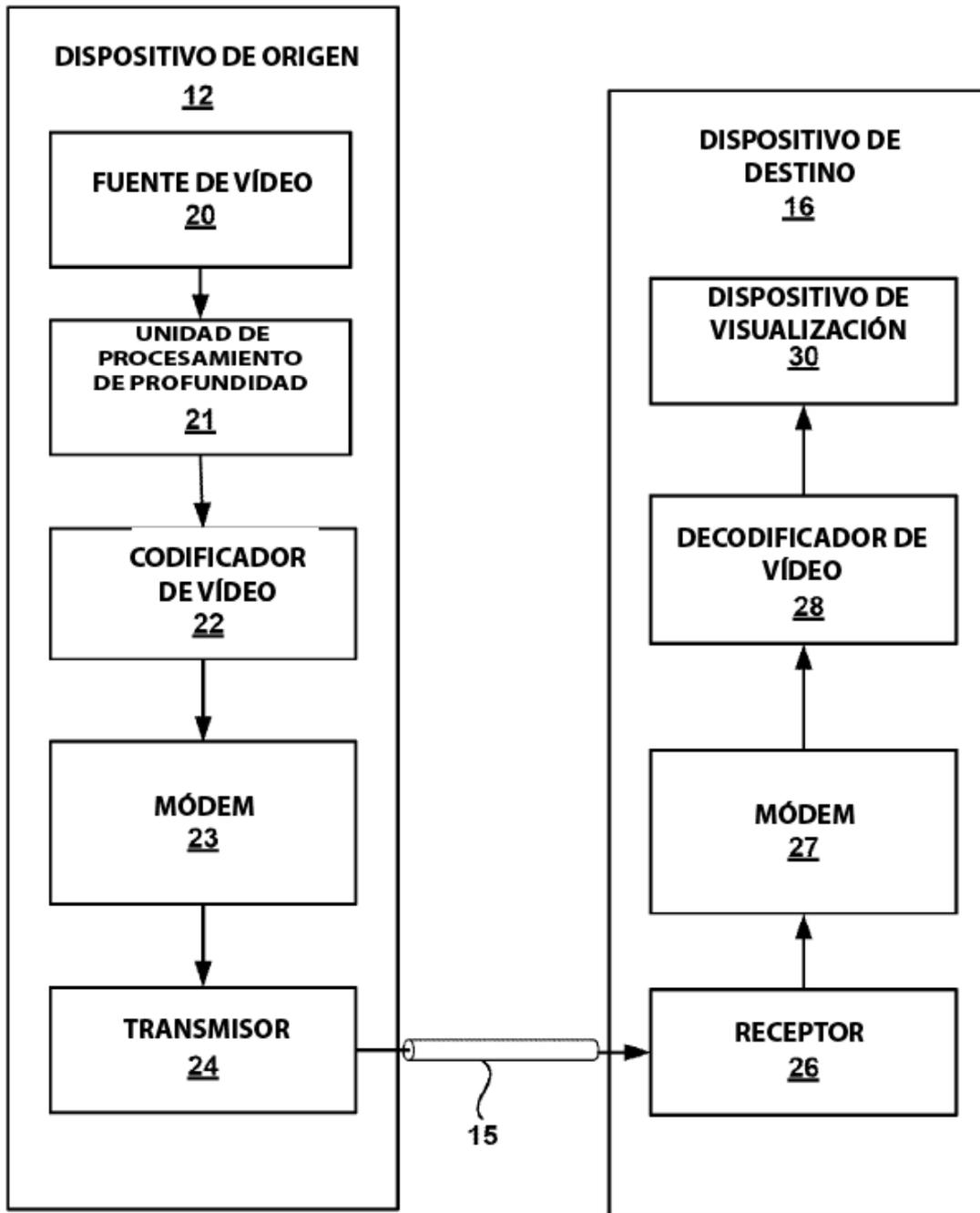


FIG. 1

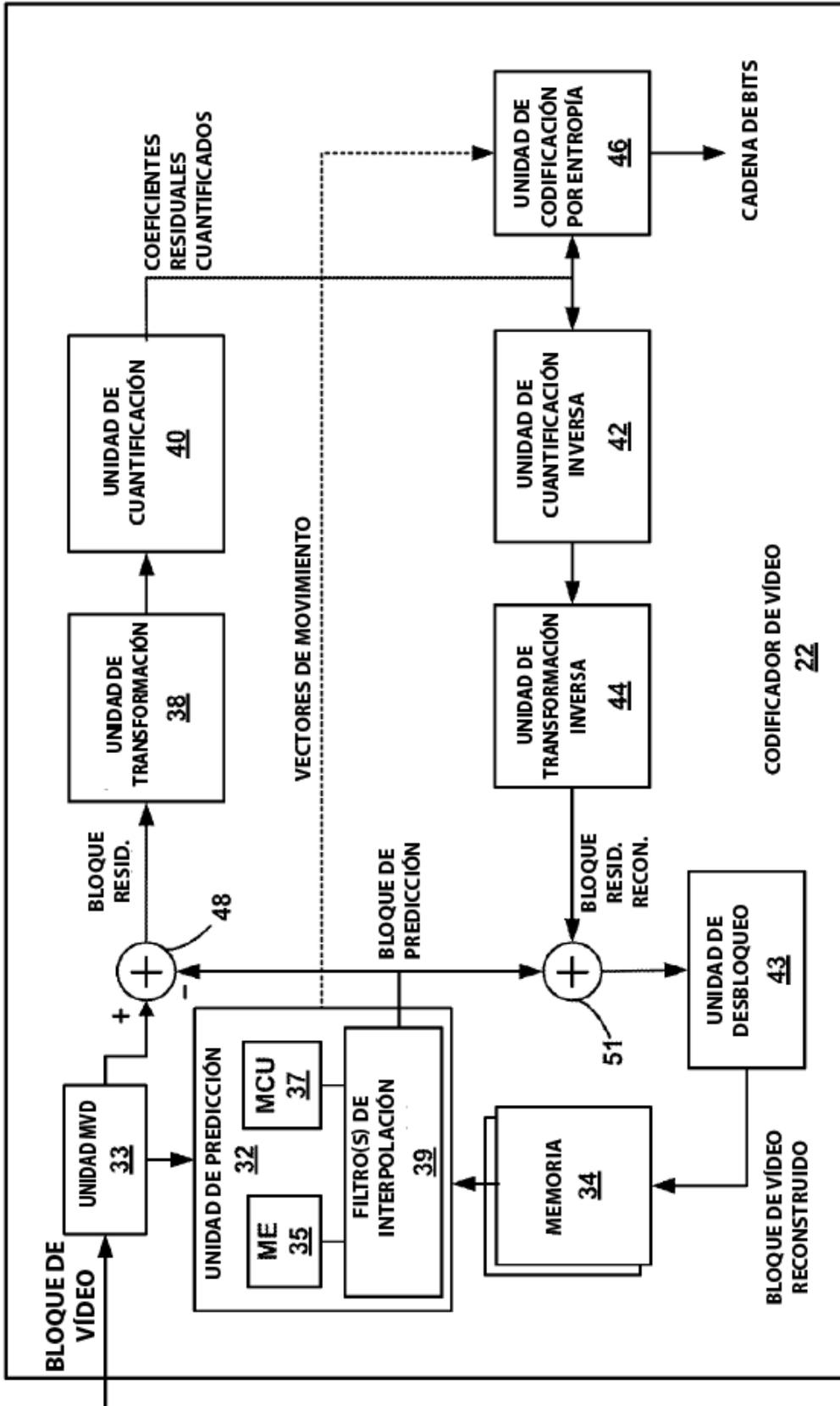


FIG. 2

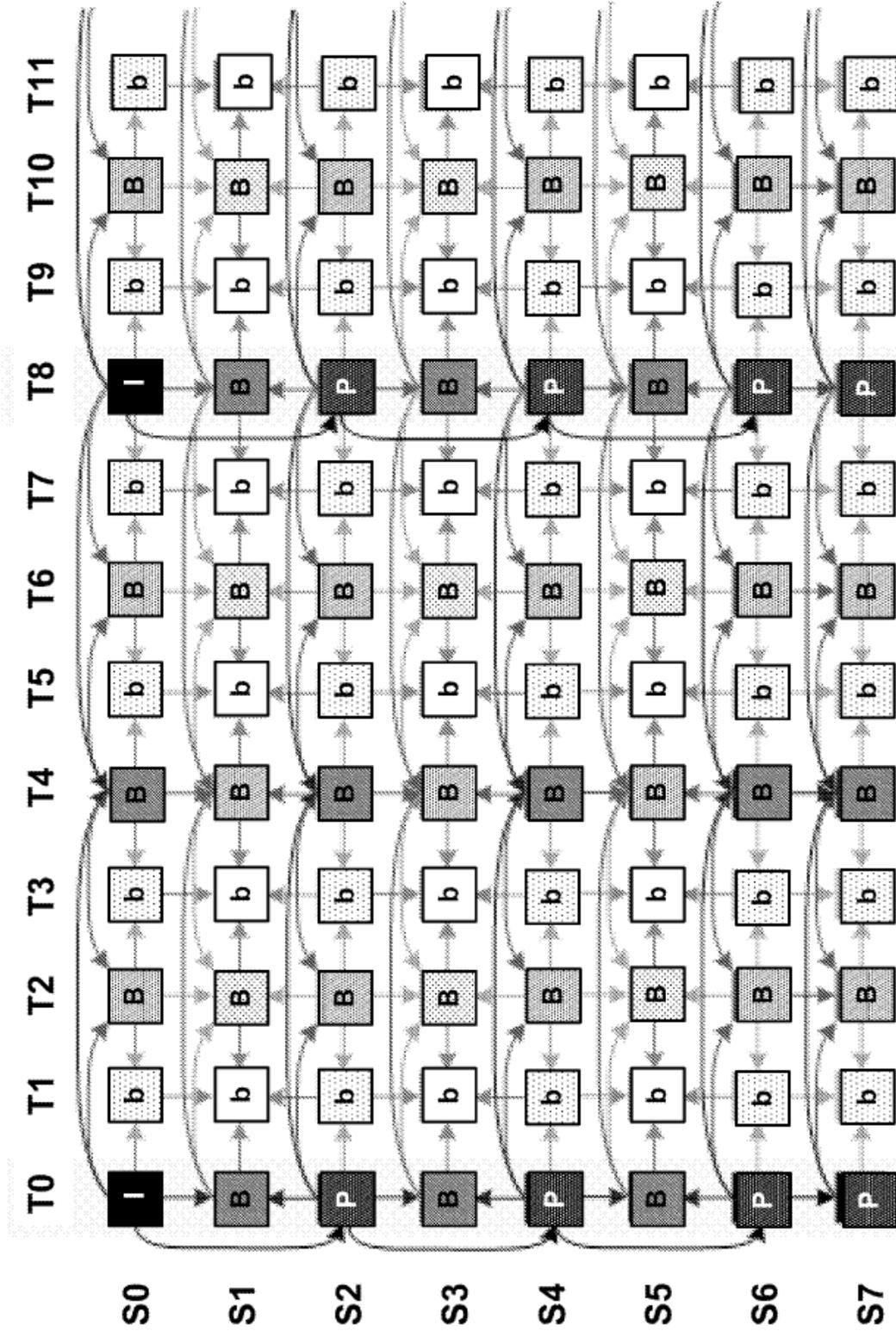


FIG. 3

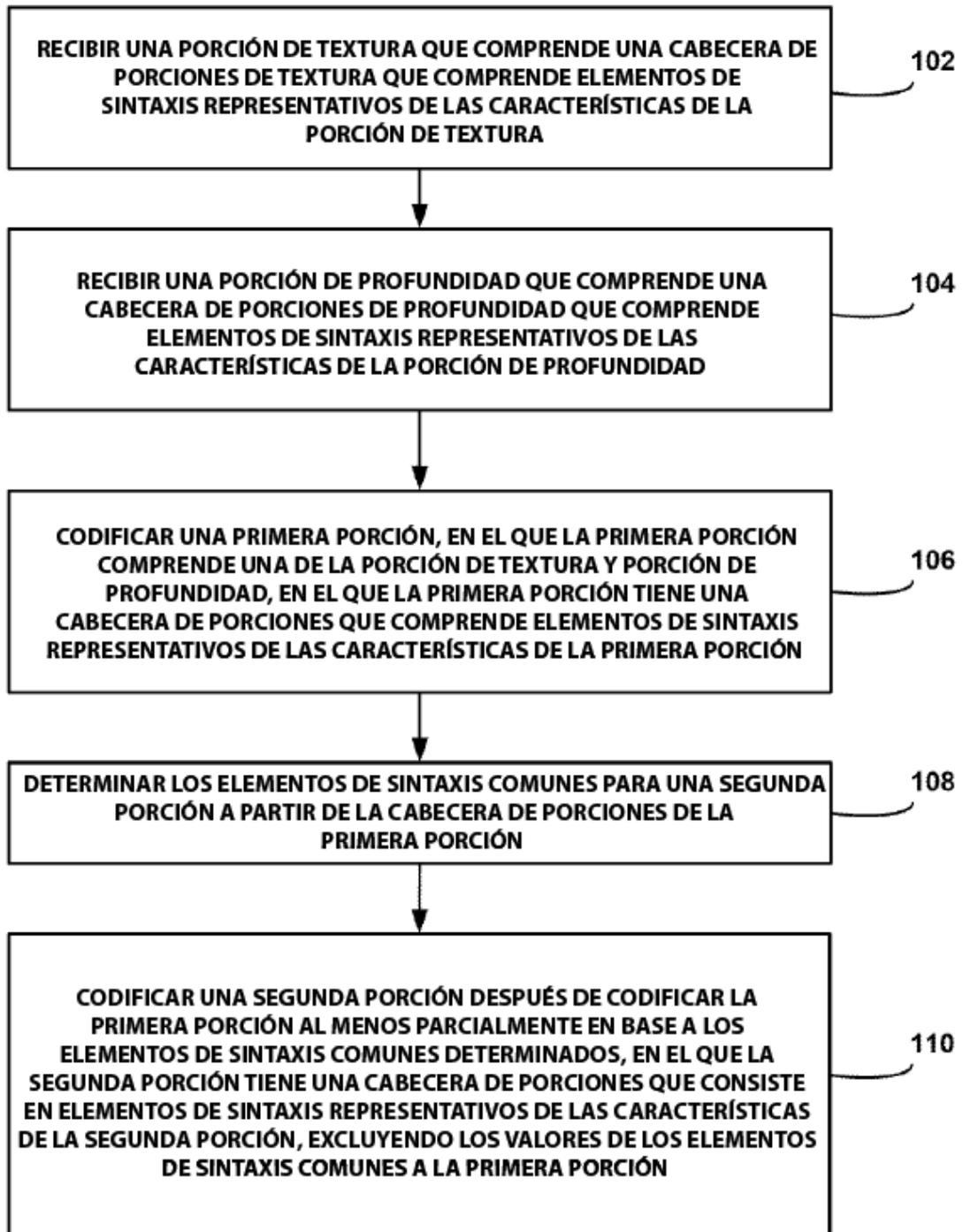


FIG. 4

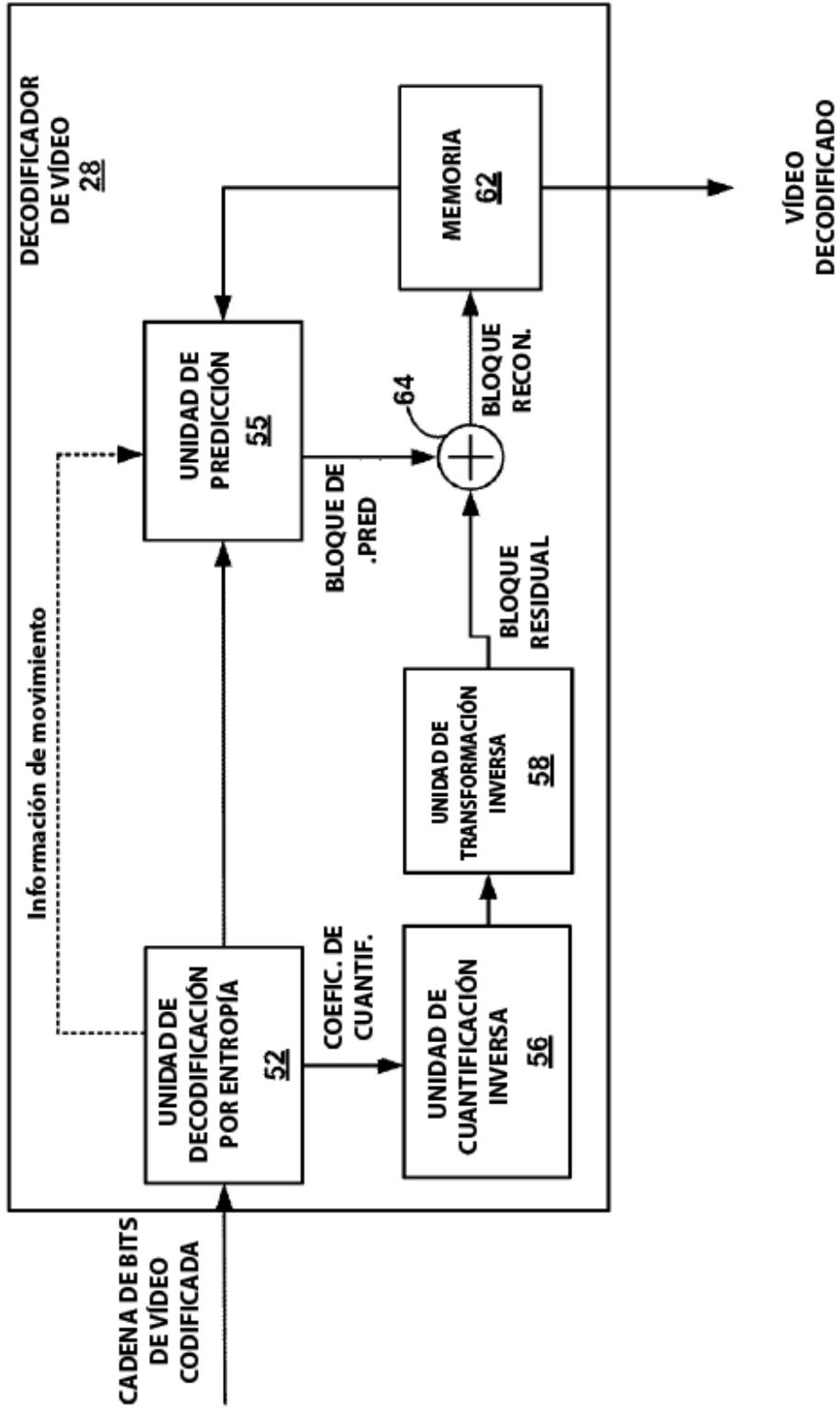


FIG. 5

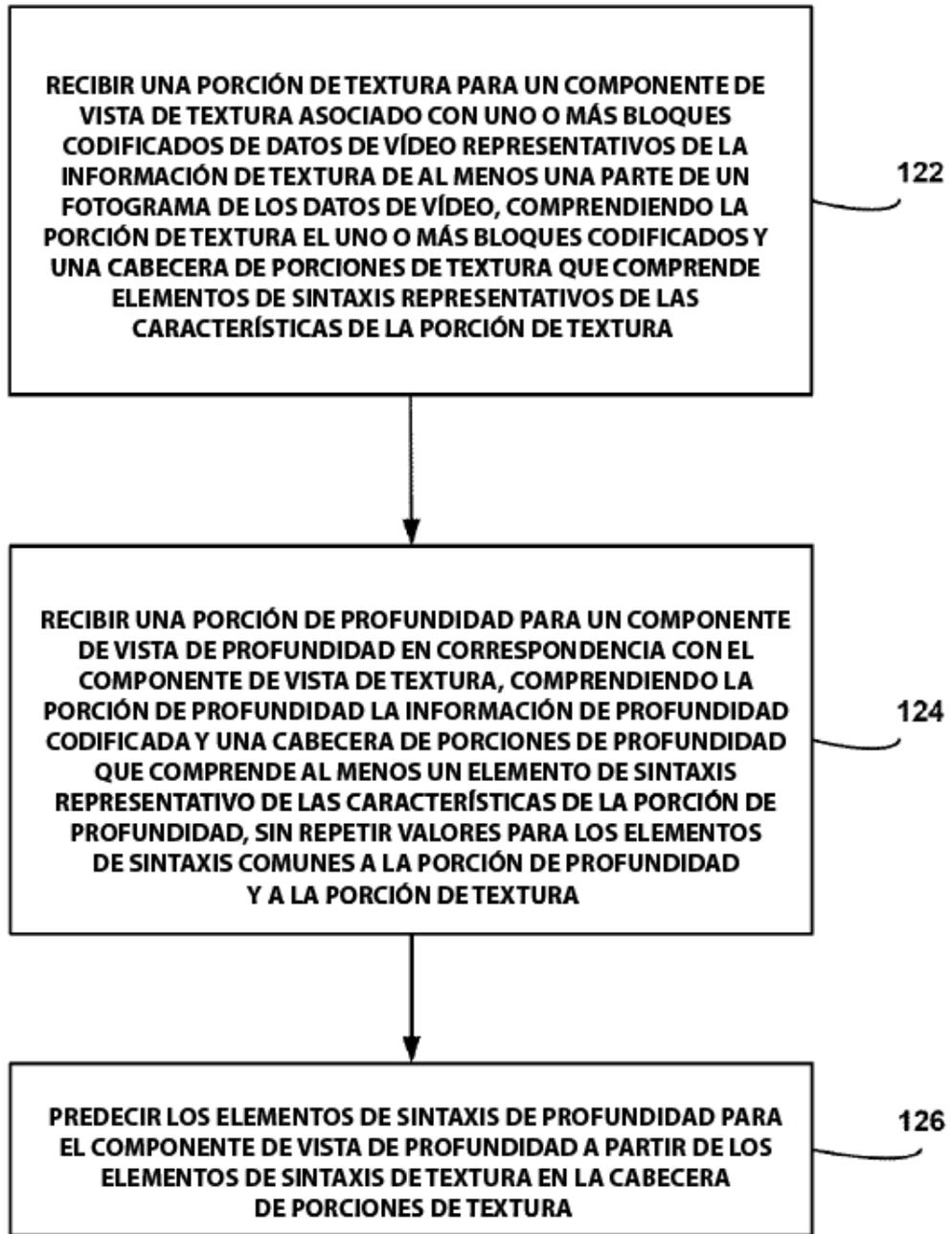


FIG. 6