

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11) Número de publicación: **2 686 936**

51) Int. Cl.:

H04N 19/172	(2014.01) H04N 13/00	(2008.01)
H04N 19/174	(2014.01) H04L 29/06	(2006.01)
H04N 19/176	(2014.01)	
H04N 19/44	(2014.01)	
H04N 19/46	(2014.01)	
H04N 19/51	(2014.01)	
H04N 19/597	(2014.01)	
H04N 19/61	(2014.01)	
H04N 19/70	(2014.01)	
H04N 19/50	(2014.01)	

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86) Fecha de presentación y número de la solicitud internacional: **20.07.2012 PCT/US2012/047701**
- 87) Fecha y número de publicación internacional: **31.01.2013 WO13016231**
- 96) Fecha de presentación y número de la solicitud europea: **20.07.2012 E 12740273 (3)**
- 97) Fecha y número de publicación de la concesión europea: **20.06.2018 EP 2735152**

54) Título: **Códec 3DVC basado en MVC que soporta el modo de predicción de movimiento de visualización interna (IVMP)**

30) Prioridad:

22.07.2011 US 201161510738 P
11.08.2011 US 201161522584 P
11.08.2011 US 201161522559 P
18.11.2011 US 201161561800 P
26.11.2011 US 201161563772 P
26.11.2011 US 201161563771 P
13.04.2012 US 201261624031 P
19.07.2012 US 201213553315

45) Fecha de publicación y mención en BOPI de la traducción de la patente:
22.10.2018

73) Titular/es:

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

72) Inventor/es:

CHEN, YING;
ZHANG, LI y
KARCZEWICZ, MARTA

74) Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 686 936 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

Códec 3DVC basado en MVC que soporta el modo de predicción de movimiento de visualización interna (IVMP)

5 **[0001]** Esta aplicación reivindica el beneficio de:

La Solicitud de Patente Provisional de Estados Unidos 61/561,800, presentada el 18 de noviembre de 2011;
 La Solicitud de Patente Provisional de Estados Unidos 61/563,771, presentada el 26 de noviembre de 2011;
 La Solicitud de Patente Provisional de Estados Unidos 61/522,559, presentada el 11 de agosto de 2011; y
 10 La Solicitud de Patente Provisional de Estados Unidos 61/510,738, presentada el 22 de julio de 2011;
 La Solicitud de Patente Provisional de Estados Unidos 61/522,584, presentada el 11 de agosto de 2011,
 La Solicitud de Patente Provisional de Estados Unidos 61/563,772, presentada el 26 de noviembre de 2011, y
 La Solicitud de Patente Provisional de Estados Unidos 61/624,031, presentada el 13 de abril de 2012.

15 **CAMPO TÉCNICO**

[0002] Esta divulgación se refiere a la codificación de vídeo tridimensional (3D).

ANTECEDENTES

20 **[0003]** Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión directa digital, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores tipo tablet, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de
 25 videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), la norma de Codificación de Vídeo de Alta Eficacia (HEVC) actualmente en desarrollo y las ampliaciones de
 30 dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de compresión de vídeo.

[0004] Las técnicas de compresión de vídeo realizan la predicción espacial (intra-imagen) y/o la predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la
 35 codificación de vídeo basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse bloques de árbol, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intra-codificado (I) de una imagen se codifican mediante predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen. Los bloques de vídeo en un fragmento inter-codificado (P o B) de una imagen
 40 pueden usar predicción espacial con respecto a muestras de referencia en bloques contiguos en la misma imagen, o predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

[0005] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los
 45 datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque inter-codificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intra-codificado se codifica de acuerdo con un modo de intra-codificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de los
 50 píxeles al dominio de las transformadas, dando como resultado unos coeficientes de transformada residuales, que posteriormente se pueden cuantificar. Los coeficientes de transformada cuantificados, dispuestos inicialmente en una matriz bidimensional, pueden explorarse con el fin de generar un vector unidimensional de coeficientes de transformada, y puede aplicarse la codificación por entropía para lograr aún más compresión.

55 **[0006]** El vídeo tridimensional (3D) es muy conveniente para una variedad de aplicaciones, pero la codificación de vídeo 3D presenta muchos desafíos.

RESUMEN

60 **[0007]** Esta divulgación describe características y técnicas aplicables a la codificación de vídeo tridimensional (3D). En un ejemplo, una técnica puede incluir codificar un bloque de vídeo de visualización de textura, y codificar un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura. La codificación del bloque de vídeo de visualización de
 65 profundidad puede incluir la codificación de un elemento sintáctico para indicar si la información de movimiento asociada con el bloque de vídeo de visualización de textura se adopta como información de movimiento asociada con el bloque de vídeo de visualización de profundidad.

[0008] Las técnicas descritas pueden corresponder a un modo de codificación al que se hace referencia en el presente documento como un modo de predicción de movimiento de visualización interna (IMVP). En este caso, un componente de visualización de profundidad (por ejemplo, el bloque de vídeo de visualización de profundidad) puede no incluir ningún valor delta adicional con respecto a su información de movimiento, y en su lugar, puede adoptar la información de movimiento de un componente de visualización de textura como su información de movimiento. Al definir un modo que adopta completamente la información de movimiento de una visualización de textura como la información de movimiento de una visualización de profundidad, sin ninguna señalización de valores delta con respecto a dicha información de movimiento, se puede lograr una compresión mejorada.

[0009] En otro ejemplo, esta divulgación describe un dispositivo que codifica datos de vídeo 3D, en el que el dispositivo comprende uno o más procesadores configurados para codificar un bloque de vídeo de visualización de textura y codifica un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura. La codificación del bloque de vídeo de visualización de profundidad incluye la codificación de un elemento sintáctico para indicar si se adopta la información de movimiento asociada con el bloque de vídeo de visualización de textura como información de movimiento asociada con el bloque de vídeo de visualización de profundidad.

[0010] En otro ejemplo, esta divulgación describe un medio de almacenamiento legible por ordenador que comprende instrucciones almacenadas en el mismo, en el que las instrucciones tras la ejecución hacen que uno o más procesadores codifiquen un bloque de vídeo de visualización de textura y codifiquen un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura. La codificación del bloque de vídeo de visualización de profundidad incluye la codificación de un elemento sintáctico para indicar si se adopta la información de movimiento asociada con el bloque de vídeo de visualización de textura como información de movimiento asociada con el bloque de vídeo de visualización de profundidad.

[0011] En otro ejemplo, esta divulgación describe un dispositivo configurado para codificar datos de vídeo 3D, comprendiendo el dispositivo medios para codificar un bloque de vídeo de visualización de textura y medios para codificar un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura, en el que el medio para codificar el bloque de vídeo de visualización de profundidad incluye medios para codificar un elemento sintáctico para indicar si la información de movimiento asociada con el bloque de vídeo de visualización de textura se adopta como información de movimiento asociada con el bloque de vídeo de visualización de profundidad.

[0012] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetivos y ventajas resultarán evidentes a partir de la descripción y los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0013]

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

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

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

La FIG. 4 es un diagrama conceptual que ilustra un orden de flujo de bits de las unidades de la capa de abstracción de red (VCL) de la capa de codificación de vídeo (NAL) dentro de una unidad de acceso.

La FIG. 5 es una ilustración conceptual de una secuencia de imágenes que forman una secuencia de vídeo, en la que un macrobloque identificado en la 4.^a imagen de la visualización de profundidad y el vector de movimiento del MB co-ubicado en la 4.^a imagen de la visualización de textura se reutilizan en el componente de visualización de profundidad.

La FIG. 6 es un diagrama conceptual que muestra una estructura de predicción que puede ser utilizada por un códec de codificación de vídeo tridimensional (3DVC).

La FIG. 7 es un diagrama conceptual que muestra una estructura de predicción del códec 3DVC que no permite la predicción entre visualizaciones para componentes de visualización de profundidad.

La FIG. 8 es un diagrama conceptual que ilustra un ejemplo de predicción entre visualizaciones asimétricas, en el que tanto la visualización izquierda (VL) como la derecha (VR) tienen una anchura media.

5 La FIG. 9 es un diagrama de flujo que ilustra una técnica que puede ser realizada por un codificador de vídeo consistente con esta divulgación.

La FIG. 10 es un diagrama de flujo que ilustra una técnica que puede ser realizada por un decodificador de vídeo consistente con esta divulgación.

10 **DESCRIPCIÓN DETALLADA**

15 **[0014]** Las técnicas de esta divulgación se refieren a la codificación de vídeo tridimensional (3D) basada en la norma ITU-T H.264/AVC y una o más extensiones que soportan la codificación de múltiples visualizaciones (MVC), como el Anexo H de la norma ITU-T H.264/AVC. Sin embargo, las técnicas también pueden aplicarse a otras normas o técnicas de codificación de vídeo, tales como la norma HEVC emergente actualmente en desarrollo, extensiones de la norma ITU-T H.264/AVC o la norma HEVC emergente, o técnicas de codificación de vídeo patentadas tales como On2 VP6/VP7/VP8.

20 **[0015]** En la codificación de vídeo 3D, a menudo hay múltiples visualizaciones diferentes que se usan colectivamente para definir una presentación de vídeo 3D. Además, cada una de las diferentes visualizaciones puede incluir tanto un componente de visualización de textura como un componente de visualización de profundidad. Los componentes de visualización de textura pueden codificarse en bloques de datos de vídeo, que se denominan "bloques de vídeo" y comúnmente se denominan "macrobloques" en el contexto H.264. De manera similar, los componentes de visualización de profundidad también se codifican como "bloques de vídeo", y comúnmente se denominan "macrobloques" en la norma H.264. Cada bloque de vídeo de textura puede tener un bloque de visualización de profundidad correspondiente. Los diferentes bloques de vídeo (textura y profundidad), sin embargo, en general se codifican por separado. Otras normas de codificación de vídeo pueden referirse a bloques de vídeo como bloques de árbol o unidades de codificación (CU).

30 **[0016]** Con la inter-codificación, los vectores de movimiento (o los valores de diferencia de vector de movimiento relativos a un predictor de vector de movimiento) se pueden usar para definir bloques predictivos, que a continuación se usan para predecir los valores de los bloques de vídeo codificados. En este caso, los llamados "valores residuales" o "valores de diferencia" se incluyen en el flujo de bits codificado, junto con los vectores de movimiento (o valores de diferencia de vector de movimiento relativos a un predictor de vector de movimiento) que identifican los bloques predictivos correspondientes. El decodificador recibe los vectores de movimiento y los valores residuales, y usa los vectores de movimiento para identificar los bloques predictivos a partir de los datos de vídeo descodificados previamente. Para reconstruir los bloques de vídeo codificados, el decodificador combina los valores residuales con los bloques predictivos correspondientes identificados por los vectores de movimiento.

40 **[0017]** Existen muchos posibles problemas con la codificación de vídeo 3D. Por ejemplo, al codificar datos de vídeo de múltiples visualizaciones, puede ser necesario resolver los siguientes problemas para hacer un códec eficiente:

- 45
1. Proporcionar la capacidad de codificación conjunta de los componentes de textura y profundidad para una o más visualizaciones;
 2. Proporcionar la capacidad de aprovechar la redundancia de movimiento entre la textura y la profundidad;
 3. Proporcionar la capacidad de transmitir los parámetros de la cámara de una manera simple y eficiente;
 4. En la adaptación de visualización, `inter_view_flag` se puede usar para descartar un componente de visualización si no pertenece a una visualización que se está utilizando para la salida. Sin embargo, en el caso 50 3DV asimétrico, una unidad de capa de abstracción de red (NAL) podría ser necesaria para la predicción de las visualizaciones con diferente resolución, incluso si este indicador es igual a 0.

[0018] Para resolver los problemas anteriores, se pueden usar varias técnicas, incluidas las siguientes:

- 55
1. Un marco para soportar la codificación conjunta de visualizaciones de profundidad y textura.
 2. Se puede usar un nuevo modo de predicción de movimiento de visualización interna (IVMP) en el nivel de macrobloque (u otro bloque de vídeo o CU) para permitir la reutilización de los vectores de movimiento entre las visualizaciones de profundidad y textura. Los aspectos de un modo IVMP se describen en detalle en esta divulgación.
 - 60 3. Los parámetros de la cámara y los rangos de profundidad pueden agregarse a un conjunto de parámetros de secuencia (SPS) o como nuevos mensajes de información de mejora suplementaria (SEI), y si dichos parámetros varían en una imagen, se puede agregar un VPS (conjunto de parámetros de visualización) o SEI.
 4. La semántica de `inter_view_flag` puede modificarse o puede definirse un nuevo elemento sintáctico en la cabecera de la unidad de capa de abstracción de red (NAL) para indicar si un componente de visualización que no es descartable para una visualización con diferente resolución también es descartable para una visualización con la misma resolución.
- 65

5. Además del `nal_unit_type` (por ejemplo, 21) para ser utilizado por el componente de visualización de profundidad, un ejemplo incluye además un nuevo `nal_unit_type` (por ejemplo, 22) para los componentes de visualización de textura que no son compatibles con H.264/MVC.

5 **[0019]** Esta divulgación puede usar las siguientes definiciones.

componente de visualización: Una *representación codificada* de una *visualización* en una *única unidad de acceso*. Cuando una visualización incluye tanto representaciones de textura como de profundidad codificadas, un componente de visualización se compone de un componente de visualización de textura y un componente de visualización de profundidad.

componente de visualización de textura: Una *representación codificada* de la textura de una visualización en una *única unidad de acceso*.

componente de visualización de profundidad: Una *representación codificada* de la profundidad de una visualización en una *única unidad de acceso*.

A las unidades de capa de abstracción de red (NAL) de capa de codificación de vídeo (VCL) codificadas en un componente de visualización de profundidad se les puede asignar `nal_unit_type` 21, como un nuevo tipo de extensión de fragmento codificado específicamente para componentes de visualización de profundidad. El componente de visualización de textura y el componente de visualización de profundidad también se pueden denominar en el presente documento un bloque de vídeo de visualización de textura y un bloque de vídeo de visualización de profundidad.

[0020] Ahora se describirá una orden de flujo de bits a modo de ejemplo. En algunos ejemplos, en cada componente de visualización, cualquier unidad NAL de fragmento codificado (con `nal_unit_type` 21) del componente de visualización de profundidad debe seguir todas las unidades NAL de fragmento codificado del componente de visualización de textura. Por simplicidad, esta divulgación puede nombrar las unidades NAL de fragmentos codificados del componente de visualización de profundidad como unidades NAL de profundidad.

[0021] Una unidad NAL de profundidad puede tener la misma estructura de cabecera de unidad NAL que la unidad NAL con `nal_unit_type` igual a 20. La FIG. 4 es un diagrama conceptual que ilustra un orden de flujo de bits de las unidades VCL NAL de los componentes de visualización dentro de una unidad de acceso.

[0022] Como se muestra en la FIG. 4, de acuerdo con esta divulgación, una unidad de acceso contiene múltiples unidades NAL con múltiples componentes de visualización. Cada componente de visualización puede consistir en un componente de visualización de textura y un componente de visualización de profundidad. El componente de visualización de textura de la visualización base, con índice de orden de visualización (`VOIdx`) igual a 0, contiene una unidad NAL de prefijo (con tipo de unidad NAL igual a 14) y una o más unidades AVC VCL NAL (con tipo de unidad NAL igual a, p. ej., 1 o 5). Los componentes de visualización de textura en otras visualizaciones contienen solo las unidades MVC VCL NAL (con un tipo de unidad NAL igual a 20). Tanto en la visualización base como en las visualizaciones que no son base, los componentes de visualización de profundidad contienen las unidades NAL de profundidad con tipo de unidad NAL igual a 21. En cualquier componente de visualización, las unidades NAL de profundidad siguen a las unidades NAL del componente de visualización de textura en orden de decodificación/flujo de bits.

[0023] Dado que el componente de visualización de textura y su componente de visualización de profundidad asociado tienen una silueta de objeto similar, comúnmente tienen un movimiento y límite de objeto similares. Por lo tanto, hay redundancia en sus campos de movimiento. Un bloque de visualización de textura y un bloque de visualización de profundidad pueden estar "asociados" si existen en la misma unidad NAL y/o si corresponden a una misma (o superpuesta) instancia espacial y/o temporal de datos de vídeo 3D. Las técnicas de esta divulgación pueden explotar esta redundancia en gran medida permitiendo un modo en el que el componente de visualización de profundidad adopta completamente la información de movimiento del componente de visualización de textura asociado, de una manera similar al denominado modo de "fusión". En este caso, el componente de visualización de profundidad puede no incluir ningún valor delta adicional con respecto a su información de movimiento, y en su lugar, puede adoptar la información de movimiento del componente de visualización de textura como su información de movimiento. Al definir un modo que adopta completamente la información de movimiento de una visualización de textura como la información de movimiento de una visualización de profundidad, sin ninguna señalización de valores delta con respecto a dicha información de movimiento, se puede lograr una compresión mejorada.

[0024] En particular, la predicción de movimiento desde un componente de visualización de textura hasta el componente de visualización de profundidad asociado puede habilitarse de acuerdo con un nuevo modo que combina la información de movimiento de la visualización de textura como la de la visualización de profundidad. En algunos ejemplos, este modo denominado de predicción de movimiento de visualización interna (IVMP) solo se puede habilitar para MB inter-codificados con componentes de visualización de profundidad. En el modo IVMP, la información de movimiento, incluidos `mb_type`, `sub_mb_type`, índices de referencia y vectores de movimiento del MB

co-ubicado en el componente de visualización de textura, es reutilizada por el componente de visualización de profundidad de la misma visualización. Se puede señalar un indicador en cada MB para indicar si usa el modo IVMP. En otras palabras, el indicador puede definirse a nivel de bloque de vídeo, por ejemplo, el nivel de macrobloque. El indicador puede estar incluido con bloques de vídeo de profundidad (por ejemplo, macrobloques de profundidad). Como se muestra en la FIG. 5, el indicador puede ser verdadero para el MB identificado en la 4.^a imagen de la visualización de profundidad y el vector de movimiento del MB que se encuentra en la 4.^a imagen de la visualización de textura (identificada como la 4.^a imagen) es reutilizado para el MB resaltado en el componente de visualización de profundidad. Tenga en cuenta que, en algunos ejemplos, el modo IVMP se aplica solo a las imágenes que no son de anclaje.

[0025] De nuevo, con respecto a las técnicas que predicen un vector de movimiento para una visualización basándose en el movimiento de otra visualización, las técnicas de esta divulgación pueden lograr una mayor compresión. Por ejemplo, algunas técnicas de codificación de vídeo escalable (SVC) pueden permitir la predicción de movimiento de una visualización de mejora basada en la información de movimiento de una visualización base, y en algunos casos, la visualización base puede ser una visualización de textura y la visualización de mejora puede ser una visualización de profundidad. En tales casos, sin embargo, los datos de diferencia de vector de movimiento (por ejemplo, un delta) siempre se codifican además de la información de predicción (o indicador) que indica que la visualización base se usa para predecir la visualización de mejora. Por el contrario, las técnicas de esta divulgación pueden utilizar un modo IVMP en el que no se codifica ni se permite ninguna información delta (por ejemplo, ningún valor de diferencia de vector de movimiento). En cambio, con el modo IVMP, la información de movimiento de la visualización de textura se adopta como la información de movimiento de la visualización de profundidad.

[0026] Cuando se adopta la información de movimiento de la visualización de textura como la información de movimiento de la visualización de profundidad, el descodificador puede usar la información de movimiento de la visualización de textura (por ejemplo, un bloque de textura) para descodificar la visualización de profundidad (por ejemplo, un bloque de profundidad correspondiente) sin recibir ni descodificar ninguna otra información de movimiento para la visualización de profundidad. En particular, el descodificador se puede configurar para interpretar un indicador de IVMP de esta manera. Por lo tanto, la información de movimiento puede excluirse de un bloque de vídeo de profundidad cuando el indicador IVMP está habilitado, y el descodificador puede configurarse para saber que el indicador IVMP habilitado significa que la información de movimiento para el bloque de vídeo de profundidad puede obtenerse del bloque de vídeo de textura correspondiente.

[0027] Un codificador coherente con esta divulgación en general puede ajustarse al esquema de codificador de codificación de vídeo de múltiples visualizaciones conjunta (JMVC), en el que las visualizaciones se codifican una a una. Dentro de cada visualización, la secuencia de textura se codifica primero, y la secuencia de profundidad se codifica a continuación.

[0028] Cuando el modo IVMP está habilitado, durante la codificación del componente de visualización de textura, el campo de movimiento de cada componente de visualización de textura se escribe en un archivo de movimiento, cuyo nombre se puede especificar en un archivo de configuración. Al codificar la secuencia de profundidad asociada de la misma visualización, el archivo de movimiento se puede leer para referencia.

[0029] El descodificador puede ser similar a un descodificador JMVC, en algunos aspectos, con la modificación de también la descodificación y la emisión de una secuencia de profundidad para cada visualización. Cuando el modo IVMP está habilitado, el movimiento de cada componente de visualización de textura se almacena y adopta como el movimiento de cada visualización de profundidad correspondiente. Para cualquier bloque en el que el modo IVMP esté deshabilitado, la visualización de profundidad puede incluir su propia información de movimiento, o puede incluir algunos otros elementos sintácticos para identificar dónde obtener, predecir y/o adoptar su respectiva información de movimiento. Sin embargo, si el modo IVMP está habilitado, la visualización de profundidad no incluye su propia información de movimiento, y la información de movimiento se obtiene mediante un descodificador a partir del componente de visualización de textura correspondiente. Por lo tanto, cuando el modo IVMP está habilitado, el bloque de vídeo de visualización de profundidad adopta la información de movimiento del bloque de vídeo de visualización de textura correspondiente, de modo que el bloque de vídeo de visualización de profundidad no incluye su propia información de movimiento.

[0030] El siguiente análisis de las FIGs. 1, 2 y 3 describe algunos ejemplos de escenarios en los que se pueden usar las técnicas 3DVC basadas en MVC de esta divulgación.

[0031] La FIG. 1 es un diagrama de bloques que ilustra un ejemplo de sistema de codificación y descodificación de vídeo 10 que puede utilizar las técnicas descritas en esta divulgación. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que genera datos de vídeo codificado que un dispositivo de destino 14 va a descodificar en un momento posterior. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera entre una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, ordenadores plegables (es decir, portátiles), ordenadores tipo tablet, descodificadores, equipos telefónicos tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de transmisión continua de vídeo o

similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0032] El dispositivo de destino 14 puede recibir los datos de vídeo codificados que se van a descodificar, mediante un enlace 16. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el enlace 16 puede comprender un medio de comunicación para permitir al dispositivo de origen 12 transmitir los datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o alámbrica, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global tal como Internet. El medio de comunicación puede incluir routers, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0033] De forma alternativa, los datos codificados pueden ser emitidos desde la interfaz de salida 22 a un dispositivo de almacenamiento 32. De forma similar, se puede acceder a los datos codificados desde el dispositivo de almacenamiento 32 mediante una interfaz de entrada. El dispositivo de almacenamiento 32 puede incluir cualquiera de una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco duro, unos discos Blu-ray, discos DVD, discos CD-ROM, una memoria flash, memoria volátil o no volátil o cualquier otro medio de almacenamiento digital adecuado, para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de almacenamiento 32 puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que pueda retener el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a los datos de vídeo almacenados desde el dispositivo de almacenamiento 32 a través de transmisión en continuo o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Ejemplos de servidores de archivos incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectado en red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados mediante cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión alámbrica (por ejemplo, DSL, módem de cable, etc.), o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento 32 puede ser una transmisión en continuo, una transmisión de descarga o una combinación de ambas.

[0034] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, en soporte de cualquiera de una diversidad de aplicaciones de multimedia, tales como radiodifusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, mediante Internet, codificación de vídeo digital para su almacenamiento en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede configurarse para soportar la transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0035] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. En el dispositivo de origen 12, la fuente de vídeo 18 puede incluir una fuente tal como un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente capturado, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de tales fuentes. Como un ejemplo, si la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas.

[0036] El vídeo capturado, pre-capturado o generado por ordenador puede ser codificado por el codificador de vídeo 12. Los datos de vídeo codificados pueden ser transmitidos directamente al dispositivo de destino 14 mediante la interfaz de salida 22 del dispositivo de origen 20. Los datos de vídeo codificados también (o de forma alternativa) pueden almacenarse en el dispositivo de almacenamiento 32 para un acceso posterior por el dispositivo de destino 14 u otros dispositivos, para su descodificación y/o reproducción.

[0037] El dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 31. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 recibe los datos de vídeo codificados por el enlace 16. Los

datos de vídeo codificados, comunicados por el enlace 16, o proporcionados en el dispositivo de almacenamiento 32, pueden incluir una diversidad de elementos sintácticos generados por el codificador de vídeo 20, para su uso por un descodificador de vídeo, tal como el descodificador de vídeo 30, en la descodificación de los datos de vídeo. Dichos elementos sintácticos pueden incluirse con los datos de vídeo codificados, transmitidos en un medio de comunicación, almacenados en un medio de almacenamiento o almacenados en un servidor de archivos.

[0038] El dispositivo de visualización 31 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 31 visualiza los datos de vídeo descodificados ante un usuario y puede comprender cualquiera de una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0039] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma de Codificación de Vídeo de Alta Eficacia (HEVC), actualmente en fase de elaboración, y pueden conformarse al Modelo de Prueba HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas privadas o industriales, tales como la norma ITU-T H.264, de forma alternativa denominada MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e ITU-T H.263. Las técnicas de codificación patentadas, tales como las denominadas On2 VP6/VP7/VP8, también pueden implementar una o más de las técnicas descritas en el presente documento.

[0040] Aunque no se muestra en la FIG. 1, en algunos aspectos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar integrados, cada uno de ellos, en un codificador y descodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos diferentes. Si procede, en algunos ejemplos, las unidades MUX-DEMUX pueden conformarse al protocolo de multiplexado ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0041] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden implementarse como cualquiera de entre una variedad de circuitos adecuados de codificadores, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables de campo (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio, legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Cada uno del codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o descodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

[0042] El equipo JCT-VC está trabajando en el desarrollo de la norma HEVC. Los esfuerzos de normalización de HEVC se basan en un modelo en evolución de un dispositivo de codificación de vídeo denominado modelo de prueba de HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a dispositivos existentes de acuerdo, por ejemplo, a la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación de intra-predicción, el HM puede proporcionar hasta treinta y tres modos de codificación de intra-predicción.

[0043] En general, el modelo de funcionamiento del HM describe que una trama o imagen de vídeo puede dividirse en una secuencia de bloques arbolados o unidades de codificación de máximo tamaño (LCU), que incluyen muestras tanto de luma como de croma. Un bloque arbolado tiene un fin similar al de un macrobloque de la norma H.264. Un fragmento incluye un cierto número de bloques arbolados consecutivos en orden de codificación. Una trama o imagen de vídeo puede dividirse en uno o más fragmentos. Cada bloque arbolado puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuádruple. Por ejemplo, un bloque arbolado, como un nodo raíz del árbol cuaternario, puede dividirse en cuatro nodos hijo, y cada nodo hijo puede, a su vez, ser un nodo padre y dividirse en otros cuatro nodos hijo. Un nodo hijo final, no dividido, como nodo hoja del árbol cuaternario, comprende un nodo de codificación, es decir, un bloque de vídeo codificado. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces que puede dividirse un bloque arbolado, y también pueden definir un tamaño mínimo de los nodos de codificación. Los bloques arbolados se pueden denominar LCU en algunos ejemplos.

[0044] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformada (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe ser de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque arbolado, con un máximo de 64x64 píxeles o más. Cada CU puede contener una o más PU y una o más TU. Los datos

sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Los modos de división pueden diferir basándose en si la CU está codificada en modo de salto o directo, codificada en modo de intra-predicción o codificada en modo de inter-predicción. Las PU pueden dividirse para tener forma no cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuádruple. Una TU puede tener forma cuadrada o no cuadrada.

[0045] La norma HEVC soporta transformadas de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU típicamente se basa en el tamaño de las PU de una CU dada, definida para una LCU dividida, aunque puede que no siempre sea así. Las TU son típicamente del mismo tamaño o de un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas mediante una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformada (TU). Los valores de diferencias de píxeles, asociados a las TU, pueden transformarse para generar coeficientes de transformada, que pueden cuantificarse.

[0046] En general, una PU incluye datos relacionados con el proceso de predicción. Por ejemplo, cuando la PU está codificada de manera intramodal, la PU puede incluir datos que describan un modo de intra-predicción para la PU. Como otro ejemplo, cuando la PU está codificada de modo de inter, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, la Lista 0, la Lista 1 o la Lista C) para el vector de movimiento.

[0047] En general, se usa una TU para los procesos de transformada y cuantificación. Una CU dada que presenta una o más PU también puede incluir una o más unidades de transformada (TU). Tras la predicción, el codificador de vídeo 20 puede calcular los valores residuales correspondientes a la PU. Los valores residuales comprenden valores de diferencias de píxeles que se pueden transformar en coeficientes de transformada, cuantificar y escanear mediante las TU, para generar coeficientes de transformada en serie para la codificación por entropía. Esta divulgación usa típicamente el término "bloque de vídeo" para referirse a un nodo de codificación de una CU. En algunos casos específicos, esta divulgación también puede usar el término "bloque de vídeo" para referirse a un bloque arbolado, es decir, una LCU o una CU, que incluye un nodo de codificación y unas PU y TU.

[0048] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, en una cabecera de una o más de las imágenes o en otras ubicaciones, que describen un cierto número de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen un modo de codificación para el fragmento respectivo. Un codificador de vídeo 20 actúa típicamente sobre bloques de vídeo dentro de fragmentos de vídeo individuales con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden presentar tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

[0049] En un ejemplo, el HM soporta la predicción en diversos tamaños de PU. Suponiendo que el tamaño de una CU particular es $2N_x \times 2N_y$, el HM soporta intra-predicción en tamaños de PU de $2N_x \times 2N_y$ o $N_x \times N_y$ e inter-predicción en tamaños de PU simétricos de $2N_x \times 2N_y$, $2N_x \times N_y$, $N_x \times 2N_y$ o $N_x \times N_y$. El HM también soporta la división asimétrica para la inter-predicción en tamaños de PU de $2N_x \times nU$, $2N_x \times nD$, $nL \times 2N_y$ y $nR \times 2N_y$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida en el 25 % y el 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así pues, por ejemplo, "2NxnU" se refiere a una CU de tamaño $2N_x \times 2N_y$ que está dividida horizontalmente con una PU de tamaño $2N_x \times 0,5N_y$ encima y una PU de tamaño $2N_x \times 1,5N_y$ debajo.

[0050] En esta divulgación, "NxN" y "N por N" pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de tamaño 16×16 tendrá 16 píxeles en la dirección vertical ($y = 16$) y 16 píxeles en la dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ presenta, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles en un bloque pueden disponerse en filas y columnas. Además, no es necesario que los bloques presenten necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N.

[0051] Tras la codificación de intra-predicción o inter-predicción mediante las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos de píxeles en el dominio espacial (también denominado dominio de píxeles) y las TU pueden comprender coeficientes en el dominio de las transformadas tras la aplicación de una transformada, por ejemplo, una transformada de coseno discreta (DCT), una transformada de enteros, una transformada wavelet o una transformada similar desde un punto de

visualización conceptual a los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y los valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU incluyendo los datos residuales para la CU y, a continuación, transformar las TU para generar coeficientes de transformada para la CU.

[0052] Tras cualquier transformada para generar coeficientes de transformada, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que los coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando compresión adicional. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos o la totalidad de los coeficientes. Por ejemplo, un valor de n bits puede redondearse a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m .

[0053] En algunos ejemplos, el codificador de vídeo 20 puede usar un orden de exploración predefinido para explorar los coeficientes de transformada cuantificados, para producir un vector en serie que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de escanear los coeficientes de transformada cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa de acuerdo con el contexto (CAVLC), la codificación aritmética binaria adaptativa según el contexto (CABAC), la codificación aritmética binaria adaptativa según el contexto basada en la sintaxis (SBAC), la codificación por entropía por división de intervalos de probabilidad (PIPE) u otros procedimientos de codificación por entropía. El codificador de vídeo 20 también puede realizar la codificación por entropía de los elementos sintácticos asociados a los datos de vídeo codificados, para su uso por el decodificador de vídeo 30 en la decodificación de los datos de vídeo.

[0054] Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto dentro de un modelo contextual a un símbolo que se va a transmitir. El contexto puede referirse, por ejemplo, a si los valores contiguos del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo que se va a transmitir. Las palabras de código en la VLC pueden construirse de tal manera que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.

[0055] La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar las técnicas descritas en esta divulgación. El codificador de vídeo 20 puede realizar la intra-codificación y la inter-codificación de bloques de vídeo dentro de fragmentos de vídeo. La intra-codificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dada. La inter-codificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El modo intra (modo I) puede referirse a cualquiera de varios modos de compresión espacial. Los modos inter, tales como la predicción unidireccional (modo P) o la bi-predicción (modo B), pueden referirse a cualquiera de varios modos de compresión de base temporal.

[0056] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una unidad de división 35, un módulo de predicción 41, una memoria de imágenes de referencia 64, un sumador 50, un módulo de transformada 52, una unidad de cuantificación 54 y una unidad de codificación por entropía 56. El módulo de predicción 41 incluye una unidad de estimación de movimiento 42, una unidad de compensación de movimiento 44 y un módulo de intra-predicción 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además una unidad de cuantificación inversa 58, un módulo de transformada inversa 60 y un sumador 62. También puede incluirse un filtro de eliminación de bloques (no mostrado en la FIG. 2) para filtrar límites de bloque a fin de eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará típicamente la salida del sumador 62. También pueden usarse filtros de bucle adicionales (en bucle o tras el bucle), además del filtro de desbloqueo.

[0057] Como se representa en la FIG. 2, el codificador de vídeo 20 recibe datos de vídeo, y la unidad de división 35 divide los datos en bloques de vídeo. Esta división también puede incluir la división en fragmentos, elementos u otras unidades mayores, así como la división de bloques de vídeo, por ejemplo, de acuerdo con una estructura de árbol cuaternario de unas LCU y CU. El codificador de vídeo 20 ilustra, en general, los componentes que codifican bloques de vídeo de un fragmento de vídeo que se va a codificar. El fragmento puede dividirse en varios bloques de vídeo (y, posiblemente, en conjuntos de bloques de vídeo denominados elementos). El módulo de predicción 41 puede seleccionar una entre una pluralidad de posibles modos de codificación, tal como una entre una pluralidad de modos de intra-codificación, o una entre una pluralidad de modos de inter-codificación, para el bloque de vídeo actual, basándose en resultados de errores (por ejemplo, la velocidad de codificación y el nivel de distorsión). El módulo de predicción 41 puede proporcionar el bloque intra-codificado o inter-codificado resultante al sumador 50 para generar datos de bloques residuales, y al sumador 62 para reconstruir el bloque codificado para su uso como una imagen de referencia.

[0058] El módulo de intra-predicción 46, dentro del módulo de procesamiento de predicción 41, puede realizar la codificación intra-predictiva del bloque de vídeo actual con respecto a uno o más bloques contiguos en la misma trama o fragmento que el bloque que va a codificarse, para proporcionar compresión espacial. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, dentro del módulo de predicción 41, realizan la codificación inter-predictiva del bloque de vídeo actual con respecto a uno o más bloques predictivos en una o más imágenes de referencia, para proporcionar compresión temporal.

[0059] La unidad de estimación de movimiento 42 puede estar configurada para determinar el modo de inter-predicción para un fragmento de vídeo de acuerdo con un patrón por defecto para una secuencia de vídeo. El patrón por defecto puede designar fragmentos de vídeo de la secuencia como fragmentos P, fragmentos B o fragmentos GPB. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar altamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento de los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo de una trama o imagen de vídeo actual con respecto a un bloque predictivo de una imagen de referencia.

[0060] Un bloque predictivo es un bloque del que se descubre que se corresponde estrechamente con la PU del bloque de vídeo que se va a codificar en términos de diferencia de píxeles, que puede determinarse mediante la suma de una diferencia absoluta (SAD), suma de diferencia al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones de píxel fraccionarias de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento en relación con las posiciones de píxeles completas y las posiciones de píxeles fraccionarias, y generar un vector de movimiento con una precisión de píxel fraccionaria.

[0061] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento inter-codificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse de una primera lista de imágenes de referencia (Lista 0) o una segunda lista de imágenes de referencia (Lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación por entropía 56 y a la unidad de compensación de movimiento 44.

[0062] La compensación de movimiento, realizada por la unidad de compensación de movimiento 44, puede implicar extraer o generar el bloque predictivo basándose en el vector de movimiento determinado por la estimación de movimiento, realizando posiblemente interpolaciones hasta la precisión de subpíxel. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El codificador de vídeo 20 forma un bloque de vídeo residual restando los valores de píxeles del bloque predictivo a los valores de píxeles del bloque de vídeo actual que se está codificando, generando valores de diferencia de píxel. Los valores de diferencia de píxel forman datos residuales para el bloque, y pueden incluir componentes de diferencia de luminancia y croma. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de compensación de movimiento 44 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por el descodificador de vídeo 30 en la descodificación de los bloques de vídeo del fragmento de vídeo.

[0063] El módulo de intra-predicción 46 puede realizar la intra-predicción de un bloque actual, como alternativa a la inter-predicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, el módulo de intra-predicción 46 puede determinar un modo de intra-predicción a usar para codificar un bloque actual. En algunos ejemplos, el módulo de intra-predicción 46 puede codificar un bloque actual usando varios modos de intra-predicción, por ejemplo, durante diferentes pasadas de codificación, y el módulo de intra-predicción 46 (o el módulo de selección de modo 40, en algunos ejemplos) puede seleccionar un modo adecuado de intra-predicción a usar, entre los modos probados. Por ejemplo, el módulo de intra-predicción 46 puede calcular valores de distorsión de velocidad usando un análisis de distorsión de velocidad para los diversos modos de intra-predicción probados, y seleccionar el modo de intra-predicción que tenga las mejores características de distorsión de velocidad entre los modos probados. El análisis de distorsión de velocidad determina, en general, una magnitud de distorsión (o error) entre un bloque codificado y un bloque original no codificado que se codificó para generar el bloque codificado, así como una velocidad de transferencia de bits (es decir, un número de bits) utilizada para generar el bloque codificado. El módulo de intra-predicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intra-predicción presenta el mejor valor de distorsión de velocidad para el bloque.

[0064] En algunos casos, el módulo de predicción 41 puede seleccionar el modo IVMP para codificar uno o más bloques de vídeo de profundidad. En este caso, la información de movimiento para un bloque de vídeo de textura correspondiente puede adoptarse para un bloque de profundidad como se describe en el presente documento. El bloque de profundidad y el bloque de textura pueden codificarse en la misma unidad NAL, y un indicador IVMP puede codificarse de modo que un descodificador pueda descodificar adecuadamente el bloque de vídeo de profundidad al reutilizar la información de movimiento del bloque de vídeo de visualización de textura correspondiente.

[0065] En cualquier caso, tras seleccionar un modo de intra-predicción para un bloque, el módulo de intra-predicción 46 puede proporcionar información que indica el modo de intra-predicción seleccionada para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indica el modo de intra-predicción seleccionado de acuerdo con las técnicas de esta divulgación. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido, que pueden incluir una pluralidad de tablas de índices de modos de intra-predicción y una pluralidad de tablas de índices de modos de intra-predicción modificadas (también denominadas tablas de asignación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones del modo de intra-predicción más probable, una tabla de índices de modos de intra-predicción y una tabla de índices de modos de intra-predicción modificadas a utilizar para cada uno de los contextos.

[0066] Después de que el módulo de predicción 41 genera el bloque predictivo para el bloque de vídeo actual, ya sea mediante la inter-predicción o la intra-predicción, el codificador de vídeo 20 forma un bloque de vídeo residual restando el bloque predictivo al bloque de vídeo actual. Los datos de vídeo residual en el bloque residual pueden ser incluidos en una o más TU y aplicados al módulo de procesamiento de transformada 52. El módulo de transformada 52 transforma los datos de vídeo residual en coeficientes de transformada residual, usando una transformada, tal como una transformada de coseno discreta (DCT) o una transformada conceptualmente similar. El módulo de procesamiento de transformada 52 puede convertir los datos de vídeo residual, desde un dominio de píxeles a un dominio de transformada, tal como un dominio de frecuencia.

[0067] El módulo de procesamiento de transformadas 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformada para reducir adicionalmente la velocidad de transmisión de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos o la totalidad de los coeficientes. El grado de cuantificación puede modificarse ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que incluye los coeficientes de transformada cuantificados. De forma alternativa, la unidad de codificación por entropía 56 puede llevar a cabo la exploración.

[0068] Tras la cuantificación, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformada cuantificados. Por ejemplo, la unidad de codificación por entropía 56 puede realizar una codificación de longitud variable adaptativa según el contexto (CAVLC), una codificación aritmética binaria adaptativa según el contexto (CABAC), una codificación aritmética binaria adaptativa según el contexto basándose en la sintaxis (SBAC), una codificación por entropía por división de intervalos de probabilidad (PIPE) u otros procedimientos o técnicas de codificación por entropía. Tras la codificación por entropía realizada por la unidad de codificación por entropía 56, el flujo de bits codificado puede transmitirse al descodificador de vídeo 30, o archivar para su posterior transmisión o recuperación por el descodificador de vídeo 30. La unidad de codificación por entropía 56 también puede realizar la codificación por entropía de los vectores de movimiento y los otros elementos sintácticos para el fragmento de vídeo actual que se está codificando.

[0069] La unidad de cuantificación inversa 58 y el módulo de transformada inversa 60 aplican una cuantificación inversa y una transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, para su uso posterior como un bloque de referencia de una imagen de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia sumando el bloque residual a un bloque predictivo de una de las imágenes de referencia de una de las listas de imágenes de referencia. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxeles fraccionarios para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque predictivo con compensación de movimiento generado por la unidad de compensación de movimiento 44 para generar un bloque de referencia para su almacenamiento en la memoria de imágenes de referencia 64. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden usar el bloque de referencia como un bloque de referencia para realizar la inter-predicción de un bloque en una imagen o trama de vídeo subsiguiente.

[0070] La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo 30 que puede implementar las técnicas descritas en esta divulgación. En el ejemplo de la FIG. 3, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 80, un módulo de predicción 81, una unidad de cuantificación inversa 86, una unidad de transformada inversa 88, un sumador 90, y una memoria de imágenes de referencia 92. El módulo de predicción 81 incluye la unidad de compensación de movimiento 82 y el módulo de intra-predicción 84.

En algunos ejemplos, el descodificador de vídeo 30 puede realizar un pase de descodificación que en general es recíproco al pase de codificación descrito con respecto al codificador de vídeo 20 de la FIG. 2.

[0071] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. La unidad de descodificación por entropía 80 del descodificador de vídeo 30 realiza la descodificación por entropía del flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos sintácticos. La unidad de descodificación por entropía 80 envía los vectores de movimiento y otros elementos sintácticos al módulo de predicción 81. El descodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel del fragmento de vídeo y/o el nivel del bloque de vídeo.

[0072] Cuando el fragmento de vídeo se codifica como un fragmento intra-codificado (I), el módulo de intra-predicción 84 del módulo de predicción 81 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual, basándose en un modo de intra-predicción señalado y datos de bloques descodificados previamente de la trama o imagen actual. Cuando la trama de vídeo es codificada como un fragmento inter-codificado (es decir, B, P o GPB), la unidad de compensación de movimiento 82 del módulo de predicción 81 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual, basándose en los vectores de movimiento y otros elementos sintácticos recibidos desde la unidad de descodificación por entropía 80. Los bloques predictivos pueden generarse a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por defecto basándose en las imágenes de referencia almacenadas en la memoria de imágenes de referencia 92.

[0073] La unidad de compensación de movimiento 82 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual analizando los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos del bloque de vídeo actual que se está descodificando. Por ejemplo, la unidad de compensación de movimiento 82 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intra-predicción o inter-predicción) usado para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de inter-predicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia del fragmento, vectores de movimiento para cada bloque de vídeo inter-codificado del fragmento, el estado de inter-predicción para cada bloque de vídeo inter-codificado del fragmento y otra información para descodificar los bloques de vídeo en el fragmento de vídeo actual.

[0074] En algunos casos, el módulo de predicción 81 puede interpretar un indicador en una unidad NAL y seleccionar el modo IVMP para descodificar uno o más bloques de vídeo de profundidad de la unidad NAL. En este caso, la información de movimiento para un bloque de vídeo de textura correspondiente puede adoptarse para un bloque de profundidad como se describe en el presente documento. El bloque de profundidad y el bloque de textura pueden codificarse en la misma unidad NAL, y un indicador IVMP puede descodificarse a partir del flujo de bits para que el descodificador de vídeo 30 pueda descodificar adecuadamente el bloque de vídeo de profundidad reutilizando la información de movimiento del bloque de vídeo de visualización de textura correspondiente.

[0075] La unidad de compensación de movimiento 82 también puede realizar la interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 82 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo para calcular valores interpolados para píxeles fraccionarios de bloques de referencia. En este caso, la unidad de compensación de movimiento 82 puede determinar los filtros de interpolación utilizados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y utilizar los filtros de interpolación para generar bloques predictivos.

[0076] La unidad de cuantificación inversa 86 cuantifica de manera inversa, es decir, descuantifica, los coeficientes de transformada cuantificados proporcionados en el flujo de bits y descodificados por la unidad de descodificación por entropía 80. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el codificador de vídeo 20 para cada bloque de vídeo en el fragmento de vídeo, para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que debería aplicarse. El módulo de transformada inversa 88 aplica una transformada inversa, por ejemplo, una DCT inversa, una transformada inversa entera o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada, con el fin de producir bloques residuales en el dominio de los píxeles.

[0077] Después de que el módulo de predicción 81 genera el bloque predictivo para el bloque de vídeo actual basado en inter o intra-predicción, el descodificador de vídeo 30 forma un bloque de vídeo descodificado al sumar los bloques residuales del módulo de transformada inversa 88 con los bloques predictivos correspondientes generados por el módulo de predicción 81. El sumador 90 representa el componente o los componentes que llevan a cabo esta operación de suma. Si se desea, también puede aplicarse un filtro de desbloqueo para filtrar los bloques descodificados con el fin de eliminar distorsiones de efecto pixelado. También pueden utilizarse otros filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) para suavizar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo descodificados en una trama o imagen dada son a

continuación almacenados en la memoria de imágenes de referencia 92, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La memoria de imágenes de referencia 92 almacena también vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 31 de la FIG. 100.

5 [0078] Para la codificación de vídeo 3D, el componente de visualización de textura y su componente de visualización de profundidad asociado pueden tener una silueta de objeto similar, y estos componentes de visualización diferentes pueden tener un límite y movimiento de objeto similar. Por lo tanto, hay redundancia en los campos de movimiento de los componentes de visualización de textura y componentes de visualización de profundidad asociados. Las técnicas de esta divulgación pueden explotar esta redundancia en mayor medida que las técnicas convencionales permitiendo un modo en el que el componente de visualización de profundidad adopta completamente la información de movimiento del componente de visualización de textura, de manera similar al denominado modo "fusión". En este caso, el componente de visualización de profundidad puede no incluir ningún valor delta adicional con respecto a su información de movimiento (es decir, puede no incluir ningún valor de diferencia de vector de movimiento) y, en su lugar, puede adoptar la información de movimiento del componente de visualización de textura como su información de movimiento.

20 [0079] En particular, la predicción de movimiento desde un componente de visualización de textura hasta el componente de visualización de profundidad asociado puede habilitarse de acuerdo con un nuevo modo que combina la información de movimiento de la visualización de textura como la de la visualización de profundidad. En algunos ejemplos, este modo denominado IVMP solo se puede habilitar para MB inter-codificados con componentes de visualización de profundidad. En el modo IVMP, la información de movimiento, incluidos `mb_type`, `sub_mb_type`, los índices de referencia y los vectores de movimiento del MB co-ubicado en el componente de visualización de textura, es reutilizada por el componente de visualización de profundidad de la misma visualización. Se puede señalar un indicador en cada MB para indicar si usa el modo IVMP. Como se muestra en la FIG. 5, el indicador puede ser verdadero para el MB identificado en la 4.^a imagen de la visualización de profundidad y el vector de movimiento del MB co-ubicado que se encuentra en la 4.^a imagen de la visualización de textura (identificada como la 4.^a imagen) es reutilizado para el MB resaltado en el componente de visualización de profundidad. Tenga en cuenta que, en algunos ejemplos, el modo IVMP se aplica solo a las imágenes que no son de anclaje. El término "imagen de anclaje" puede definirse como cualquier punto de acceso aleatorio (RAP) que es diferente de una imagen de refresco de descodificación instantánea (IDR).

35 [0080] Como se mencionó anteriormente, en relación con las técnicas convencionales que predicen un vector de movimiento para una visualización basándose en el movimiento de otra visualización, las técnicas de esta divulgación pueden lograr una mayor compresión. Por ejemplo, algunas técnicas escalables convencionales pueden permitir la predicción de movimiento de una visualización de mejora basándose en la información de movimiento de una visualización base, y en algunos casos, la visualización base puede ser una visualización de textura y la visualización de mejora puede ser una visualización de profundidad. En tales casos, sin embargo, un valor de diferencia de vector de movimiento (por ejemplo, un delta) siempre se codifica además de la información de predicción (o indicador) que indica que la visualización base se usa para predecir la visualización de mejora. Por el contrario, las técnicas de esta divulgación pueden utilizar un modo IVMP en el que no se codifica ni permite ninguna información delta. En cambio, con el modo IVMP, la información de movimiento de la visualización de textura se adopta como la información de movimiento de la visualización de profundidad.

45 [0081] Se describirán ahora detalles adicionales de varias técnicas de señalización para señalar datos de vídeo comprimidos. Un conjunto de parámetros de visualización (VPS) puede señalarse como "en banda", lo cual significa que el conjunto de parámetros está asociado con las imágenes codificadas y se transmiten juntos en un canal o sesión. Un VPS, si está presente en la Unidad de acceso (AU), que es la representación codificada de una instancia de tiempo del flujo de bits, puede necesitar preceder a cualquier unidad VCL NAL. Es posible que múltiples tramas tengan VPS idénticos duplicadas para introducir la resistencia.

50 [0082] En algunos ejemplos, las técnicas de esta divulgación pueden abordar un `inver_view_flag`, y pueden extender la semántica de `inter_view_flag`. En un ejemplo, un `inter_view_flag` igual a 0 especifica que el componente de visualización actual no es utilizado para la predicción entre visualizaciones por ningún otro componente de visualización en la unidad de acceso actual con la misma o diferente resolución espacial. En este ejemplo, `inter_view_flag` igual a 1 puede especificar que el componente de visualización actual pueda ser utilizado para la predicción entre visualizaciones por otros componentes de visualización en la unidad de acceso actual.

60 [0083] El valor de `inter_view_flag` puede ser el mismo para todas las unidades VCL NAL de un componente de visualización.

65 [0084] En un ejemplo, las visualizaciones izquierda y derecha tienen la mitad de resolución y la visualización central tiene una resolución completa. En un perfil 3DV asimétrico, este indicador se puede establecer en 1, por ejemplo, para la visualización derecha. Sin embargo, si se extrae un sub-flujo de MVC, este indicador no tiene que ser 1.

[0085] Definir un indicador llamado **inter_asy_view_flag**:

nal_unit_header_mvc_extension () {	C	Descriptor
non_idr_flag	Todos	u(1)
priority_id	Todos	u(6)
view_id	Todos	u(10)
temporal_id	Todos	u(3)
anchor_pic_flag	Todos	u(1)
inter_asy_view_flag	Todos	u(1)
reserved_one_bit	Todos	u(1)
}	Todos	u(1)

5 En algunos ejemplos, **inter_asy_view_flag** igual a 0 especifica que el componente de visualización actual no es utilizado para la predicción entre visualizaciones por ningún otro componente de visualización en la unidad de acceso actual con una resolución espacial diferente. **inter_asy_view_flag** igual a 1 especifica que el componente de visualización actual puede ser utilizado para la predicción entre visualizaciones por otros componentes de visualización con una resolución espacial diferente en la unidad de acceso actual.

10 **[0086]** En el ejemplo anterior, para la visualización izquierda, las unidades NAL pueden tener **inter_view_flag** igual a 1 e **inter_asy_view_flag** igual a 1. Para la visualización derecha, las unidades NAL pueden tener **inter_view_flag** igual a 0 y **inter_asy_view_flag** igual a 1, y para la visualización central, todas las unidades NAL pueden tener estos dos indicadores igual a 0.

15 **[0087]** Esta divulgación puede proporcionar una respuesta a la convocatoria de propuestas (CfP) sobre codificación de vídeo 3D emitida por MPEG. La propuesta se basa en el software de referencia H.264/MVC JMVC con varias mejoras y adiciones, que pueden incorporar la codificación conjunta de textura y profundidad para múltiples visualizaciones. La propuesta de esta divulgación puede contener codificación conjunta de textura y profundidad, predicción desde textura a profundidad dentro de una visualización, y codificación asimétrica de componentes de visualización con diferentes resoluciones. En la propuesta, el software de síntesis de visualizaciones MPEG se puede usar para la generación de visualizaciones sin ninguna modificación.

25 **[0088]** En comparación con el anclaje de JMVC 8.3.1, para el caso de 2 visualizaciones, la propuesta de esta divulgación puede lograr una reducción de la velocidad de hasta un 22,6 % (de promedio 11,7 %) y para el caso de 3 visualizaciones, reducción de la velocidad de hasta un 15,8 % (de promedio un 7,3 %) cuando las velocidades de transmisión de bits son las velocidades de transmisión de bits totales tanto de la textura como de la profundidad de las dos visualizaciones y los valores pico de la relación señal/ruido (PSNR) son los valores medios de PSNR de las dos visualizaciones de textura descodificadas.

30 **[0089]** Para el caso de 2 visualizaciones, si se utilizan las velocidades de transmisión de bits totales frente a los valores de PSNR de la visualización sintetizada, la reducción de la velocidad de BD es de hasta un 24,7 % (y un 13,9 % de media) y para el caso de 3 visualizaciones, si se utilizan las velocidades de transmisión de bits totales respecto a los valores promedio de PSNR de las dos visualizaciones sintetizadas, la reducción de la velocidad de BD es de hasta un 19,0 % (y 15,0 % de promedio).

35 **[0090]** Esta divulgación puede proporcionar lo siguiente:

- Compatibilidad con H.264/AVC alto perfil y H.264/MVC alto perfil estéreo, y potencialmente alto perfil de múltiples visualizaciones;
- Codificación conjunta de textura y profundidad para secuencias de múltiples visualizaciones;
- Resoluciones espaciales y temporales simétricas para los componentes de visualización de textura y profundidad de cada visualización;
- Resoluciones espaciales asimétricas para diferentes visualizaciones.

45 Las modificaciones adicionales del códec en la parte superior del códec H.264/MVC también pueden incluir:

- Sintaxis de alto nivel para soportar la codificación conjunta de componentes de visualización de textura y profundidad;
- Predicción de vector de movimiento entre componentes de visualización de textura y profundidad y un modo en el que se adopta el movimiento de visualización de profundidad desde el movimiento de visualización de textura asociado.

[0091] Esta divulgación también describe otras herramientas, tales como herramientas que permiten la predicción entre componentes de visualización en diferentes resoluciones, y la predicción de las cabeceras de fragmento desde el componente de visualización de textura hasta el componente de visualización de profundidad correspondiente. El componente de visualización de textura y el componente de visualización de profundidad pueden formar un componente de visualización que es una imagen codificada de una visualización en una unidad de acceso. Por lo tanto, las técnicas pueden permitir la adopción de información de movimiento de acuerdo con el modo de IVMP descrito, o predicción (que incluye deltas) de información de movimiento para una visualización de profundidad con respecto a una visualización de textura. Ambas herramientas pueden permitir la flexibilidad de codificación, aunque la mejor compresión se puede lograr limitando las herramientas hasta cierto punto. Por ejemplo, el modo de IVMP descrito en el presente documento puede estar limitado a las imágenes que no son de anclaje.

[0092] A lo largo del documento, AVC se refiere a H.264/AVC alto perfil. Si se hace referencia a cualquier otro perfil o modificación de H.264/AVC, la modificación o el nombre del perfil se especificarán explícitamente. Por ejemplo, H.264/MVC o MVC se refiere a la extensión de múltiples visualizaciones de H264/AVC. Sin embargo, cualquier modificación o perfil del H.264/AVC pertenece a la familia AVC, por lo tanto, el códec propuesto, si es compatible con MVC alto perfil estéreo, también es compatible con AVC alto perfil estéreo.

[0093] Ahora se proporcionará una descripción del códec. En esta sección, el códec 3DVC propuesto se describe a partir de dos aspectos, un marco de alto nivel y técnicas de codificación de bajo nivel. Si es deseable definir un formato 3DV que pueda tener configuraciones de 2 visualizaciones y 3 visualizaciones correspondientes a aplicaciones potencialmente diferentes, las técnicas en el caso de 3 visualizaciones pueden formar un superconjunto de aquellas en el caso de 2 visualizaciones. Por lo tanto, en esta sección, se ilustra en primer lugar el marco de alto nivel aplicable a ambos casos, seguido de la descripción del códec de las técnicas en el caso de 2 visualizaciones que son aplicables al caso de 3 visualizaciones, y a continuación se describen las técnicas utilizadas solo en el caso de 3 visualizaciones.

[0094] Un marco de alto nivel puede usar las siguientes definiciones:

componente de visualización: Una *representación codificada* de una *visualización* en una *única unidad de acceso*. Cuando una visualización incluye tanto representaciones de textura como de profundidad codificadas, un componente de visualización se compone de un componente de visualización de textura y un componente de visualización de profundidad.

componente de visualización de textura: Una *representación codificada* de la textura de una visualización en una *única unidad de acceso*.

componente de visualización de profundidad: Una *representación codificada* de la profundidad de una visualización en una *única unidad de acceso*.

Las unidades VCL NAL codificadas en un componente de visualización de profundidad pueden asignarse con `nal_unit_type` 21, como un nuevo tipo de extensión de fragmento codificado específicamente para componentes de visualización de profundidad.

[0095] Ahora se describirá una orden de flujo de bits. En cada componente de visualización, cualquier unidad NAL de fragmento codificado (con `nal_unit_type` 21) del componente de visualización de profundidad puede necesitar seguir todas las unidades NAL de fragmento codificado del componente de visualización de textura. Para simplificar, esta divulgación nombra las unidades NAL de fragmento codificado del componente de visualización de profundidad como unidades NAL de profundidad.

[0096] Una unidad NAL de profundidad tiene la misma estructura de cabecera de unidad NAL que la unidad NAL con `nal_unit_type` igual a 20. La FIG. 4 muestra un orden a modo de ejemplo de flujo de bits de las unidades VCL NAL de los componentes de visualización dentro de una unidad de acceso.

[0097] Como se muestra en la FIG. 4, en un códec de vídeo 3D a modo de ejemplo, una unidad de acceso contiene múltiples componentes de visualización, cada uno de los cuales consiste en un componente de visualización de textura y un componente de visualización de profundidad. El componente de visualización de textura de la visualización base, con índice de orden de visualización (VOIdx) igual a 0, contiene una unidad NAL de prefijo (con tipo de unidad NAL igual a 14) y una o más unidades AVC VCL NAL (con tipo de unidad NAL igual a, p. ej., 1 o 5). Los componentes de visualización de textura en otras visualizaciones contienen solo las unidades MVC VCL NAL (con un tipo de unidad NAL igual a 20). Tanto en la visualización base como en las visualizaciones que no son base, los componentes de visualización de profundidad contienen las unidades NAL de profundidad con tipo de unidad NAL igual a 21. En cualquier componente de visualización, las unidades NAL de profundidad siguen a las unidades NAL del componente de visualización de textura en orden de descodificación/flujo de bits.

[0098] En un caso de 2 visualizaciones, esta divulgación puede adoptar una codificación de resolución media para las visualizaciones izquierda y derecha. Las características del códec propuesto pueden incluir:

- Resolución espacial con mitad horizontal o mitad vertical;
- La misma resolución para los componentes de visualización de textura y los componentes de visualización de profundidad de cada visualización;
- Visualización base de resolución media compatible con AVC alto perfil (solo textura);
- AVC alto perfil estéreo compatible con visualizaciones estereoscópicas de resolución media (solo textura);
- Predicción entre visualizaciones desde el componente de visualización de profundidad de la visualización base hasta el componente de visualización de profundidad de la visualización no de base;
- Textura para predicción de profundidad dentro de un componente de visualización.

[0099] La resolución medio espacial MVC se menciona a continuación y se menciona en la Tabla 1 a continuación. Todas las secuencias pueden codificarse con una resolución espacial media. En comparación con la codificación compatible con tramas H.264/AVC, la resolución medio espacial MVC es más eficiente y es más conveniente para cumplir con los siguientes requisitos:

- Compatibilidad hacia adelante: un flujo de bits 3DVC de 2 visualizaciones contiene un sub-flujo de bits MVC, que además contiene un sub-flujo de bits AVC. Por lo tanto, el códec propuesto cumple este requisito, especialmente: *"todos los flujos de bits comprimidos que se ajustan a este modo permitirán que los descodificadores de AVC existentes reconstruyan muestras de visualizaciones mono y estéreo a partir del flujo de bits"*.
- Compatibilidad estéreo/mono: Las unidades VCL NAL pueden extraerse simplemente comprobando el tipo de unidad NAL para obtener el flujo de bits MVC o AVC. Por lo tanto, el códec propuesto cumple este requisito, especialmente: *"El formato de datos comprimidos incluirá un modo que permita la extracción simple de flujos de bits para salidas estéreo y mono, y soporta la reconstrucción de alta fidelidad de muestras desde las visualizaciones izquierda y derecha del vídeo estéreo"*.

[0100] Las secuencias de resolución espacial media pueden obtenerse mediante el filtro de muestreo descendente MPEG 13-etapas ([2,0,-4,-3,5,19,26,19,5,-3,-4,0,2]/64) para secuencias de textura y profundidad. Para lograr una mejor calidad, el muestreo descendente se puede aplicar tanto horizontal como verticalmente. Para las secuencias con componentes de alta frecuencia horizontales dominantes, se puede usar la mitad de resolución vertical. En algunos ejemplos, se considera que solo una secuencia pertenece a esta categoría: *"Poznan_Hall2"*. Se considera que otras secuencias tienen componentes verticales de alta frecuencia dominantes, y se aplica un muestreo descendente horizontal para obtener la mitad de secuencias de resolución horizontales.

[0101] Se puede usar una resolución simétrica para textura y profundidad. Un componente de visualización de profundidad puede codificarse como una secuencia mono de 8 bits con la misma resolución que el componente de visualización de textura de la misma visualización. En tal configuración, la predicción desde el componente de visualización de textura hasta el componente de visualización de profundidad se puede realizar sin escalar, por ejemplo, los píxeles o vectores de movimiento en un macrobloque (MB).

[0102] La predicción entre visualizaciones para los componentes de visualización de profundidad puede ser compatible. Un componente de visualización de profundidad puede ser predicho por otros componentes de visualización de profundidad en la misma unidad de acceso, de la misma manera que la predicción entre visualizaciones en MVC. Un componente de visualización de profundidad se refiere a un conjunto de parámetros de secuencia (SPS) de subconjunto, que tiene la dependencia de visualización señalada en la extensión SPS MVC.

[0103] Típicamente, la dependencia de predicción de los componentes de visualización de profundidad comparte la misma dependencia de visualización de los componentes de visualización de textura, como se muestra en la FIG. 6. También se observa que varias secuencias no pueden beneficiarse de la predicción entre visualizaciones entre visualizaciones de profundidad. Por lo tanto, la predicción entre visualizaciones para visualizaciones de profundidad se puede simplemente deshabilitar para dichos casos. La FIG. 6 muestra una estructura de predicción del códec 3DVC. Los componentes de visualización de profundidad (mostrados con rayado cruzado) tienen la misma estructura de predicción que los componentes de visualización de textura (se muestran sin sombreado).

[0104] Por lo tanto, un indicador (`disable_depth_inter_view_flag`) se puede señalar en SPS para deshabilitar o habilitar la predicción entre visualizaciones para visualizaciones de profundidad. A continuación se describe más detalladamente el diseño de SPS para casos de 2 visualizaciones y 3 visualizaciones. Para las secuencias de mapas de profundidad que pueden beneficiarse de la predicción entre visualizaciones, los componentes de visualización de profundidad tienen las mismas estructuras de predicción entre visualizaciones e inter-predicción que los componentes de visualización de textura, como se muestra en la FIG. 6.

[0105] La FIG. 7 muestra una estructura de predicción del códec 3DVC que no permite la predicción entre visualizaciones para componentes de visualización de profundidad. Un componente ilustrado en la FIG. 7 sin sombreado indica una visualización de textura y el sombreado con rayado cruzado indica una visualización de

profundidad. Como se muestra en la FIG. 7, la predicción entre visualizaciones puede habilitarse para componentes de visualización de textura, pero deshabilitarse totalmente para componentes de visualización de profundidad. En tal caso, un componente de visualización de profundidad puede tener un tipo de fragmento diferente que el componente de visualización de textura correspondiente.

[0106] Ahora se describirá la predicción de movimiento desde la textura hasta la profundidad. Dado que el componente de visualización de textura y su componente de visualización de profundidad asociado tienen una silueta de objeto similar, tienen un límite y movimiento de objetos similares, por lo que existe redundancia en sus campos de movimiento.

[0107] De acuerdo con esta divulgación, la predicción de movimiento desde un componente de visualización de textura hasta el componente de visualización de profundidad asociado puede habilitarse como un nuevo modo en el códec propuesto. En algunos ejemplos, el modo de predicción de movimiento de visualización interna (IVMP) está habilitado para un MB inter-codificado solo en componentes de visualización de profundidad. En el modo IVMP, la información de movimiento, incluidos `mb_type`, `sub_mb_type`, los índices de referencia y los vectores de movimiento del MB co-ubicado en el componente de visualización de textura, es reutilizada por el componente de visualización de profundidad de la misma visualización. Se puede señalar un indicador en cada MB para indicar si usa el modo IVMP. De acuerdo con la FIG. 5, el indicador puede ser verdadero para la 4.^a imagen de la visualización de profundidad y el vector de movimiento del MB co-ubicado en la 4.^a imagen de la visualización de textura (etiquetada como 4.^a imagen) se reutiliza para el MB en el componente de visualización de profundidad. En algunos ejemplos, el modo IVMP se aplica solo a las imágenes sin anclaje.

[0108] La predicción de cabecera de fragmento se describirá ahora. Para cada componente de visualización, puede haber redundancia entre las cabeceras de fragmento del componente de visualización de profundidad y el componente de visualización de textura. Por lo tanto, dada la cabecera de fragmento de un componente de visualización de textura, el componente de visualización de profundidad dentro de la misma visualización de la misma unidad de acceso tiene la mayoría de su información de cabecera de fragmento ya determinada.

[0109] De acuerdo con esta divulgación, los componentes de visualización de profundidad comparten la mayoría de los elementos sintácticos de la cabecera de fragmento de los componentes de visualización de textura correspondientes. Los diferentes elementos sintácticos pueden incluir `pic_parameter_set_id`, `slice_qp_delta`, y potencialmente elementos sintácticos relacionados con la construcción de la lista de imágenes de referencia, incluyendo `num_ref_idx_10_active_minus1`, `num_ref_idx_11_active_minus1` y la tabla de sintaxis de modificación de lista de imágenes de referencia.

[0110] La cabecera de fragmento de un componente de visualización de profundidad puede señalarse en la extensión de profundidad de cabecera de fragmento. Tenga en cuenta que `pred_slice_header_depth_idc` se puede señalar en el conjunto de parámetros de secuencia. En algunos ejemplos, el codificador siempre puede configurarlo para que sea 1.

[0111] La sintaxis de extensión de profundidad de cabecera de fragmento a modo de ejemplo puede ajustarse al ejemplo de la Tabla 1 a continuación.

Tabla 1

slice_header_depth_extension() {	C	Descriptor
if (pred_slice_header_depth_idc==0		
slice_header ()		
en caso contrario {		
pic_parameter_set_id	2	ue(v)
if(slice_type == P slice_type == B) {		
num_ref_idx_active_override_flag	2	u(l)
if(num_ref_idx_active_override_flag) {		
num_ref_idx10_active_minus1	2	ue(v)
if(slice_type == B)		
num_ref_idx_11_active_minus1	2	ue(v)
}		
}		

slice_header_depth_extension() {	C	Descriptor
ref_pic_list_mvc_modification ()	2	
}		
slice_qp_delta	2	se(v)
}		
}		

5 [0112] Ahora se describirá un caso de 3 visualizaciones. Las técnicas de esta divulgación pueden adoptar codificación de resolución media para las visualizaciones izquierda y derecha, y resolución completa para la visualización central. Los procedimientos de codificación habilitados en el caso de 2 visualizaciones también pueden ser compatibles para el códec en el caso de 3 visualizaciones. El códec puede contener las siguientes características para el caso de 3 visualizaciones:

- Resolución espacial asimétrica en diferentes puntos de visualización;
- Predicción entre visualizaciones desde una visualización de baja resolución a una visualización de alta resolución;
- El sub-flujo de bits que contiene los componentes de visualización de textura de las visualizaciones de baja resolución es compatible con H.264/MVC alto perfil estéreo.
- La señalización de la dependencia de predicción entre visualizaciones para las visualizaciones de alta resolución.

15 [0113] Ahora se describirá la predicción entre visualizaciones en un códec 3DVC asimétrico. Se puede habilitar la predicción desde una visualización de baja resolución reconstruida a una visualización de alta resolución, tanto entre los componentes de visualización de textura como entre los componentes de visualización de profundidad.

20 [0114] Más específicamente, en el caso de 3 visualizaciones, la visualización de la izquierda y la derecha se pueden codificar con resolución media y la visualización central se puede codificar con resolución completa. Cuando ocurre la predicción entre visualizaciones desde un componente de visualización de resolución media hasta un componente de visualización de resolución completa (textura o profundidad), la imagen descodificada de un componente de visualización de resolución media, si se va a usar para la predicción entre visualizaciones, se muestrea ascendientemente con el filtro AVC de 6 etapas [1, -5, 20, 20, -5, 1]/32. En este caso, tanto la imagen de baja resolución (necesaria para la salida) como la imagen muestreada ascendientemente pueden necesitar coexistir temporalmente en la memoria intermedia. Las imágenes muestreadas ascendientemente desde las visualizaciones izquierda y derecha se pueden poner en las listas de imágenes de referencia de un componente de visualización de la visualización central en la misma unidad de acceso.

30 [0115] Una predicción entre visualizaciones asimétrica se muestra en la FIG. 8, en la que tanto la visualización izquierda (VL) como la derecha (VR) tienen una anchura media. Dado que la dependencia de visualización les permite ser utilizadas como referencias entre visualizaciones para la visualización central (VC), ambas se muestrean ascendientemente a imágenes intermedias.

35 [0116] En aras de la simplicidad, las visualizaciones de baja resolución, que son compatibles con MVC (si se considera solo la textura), se llaman visualizaciones MVC, independientemente de si una "visualización MVC" se refiere solo a la parte de textura o ambas partes de textura y profundidad. Las otras visualizaciones con resolución completa se conocen como visualizaciones adicionales. Por lo tanto, en el caso de 3 visualizaciones, son dos visualizaciones MVC y una visualización adicional. Cada visualización MVC contiene tanto la textura como la profundidad en la misma resolución que es la mitad de la resolución de la visualización adicional.

40 [0117] Ahora se describirá un diseño de conjunto de parámetros de secuencia. En algunos aspectos de esta divulgación, se puede introducir una nueva extensión SPS. La nueva extensión SPS se agrega al subconjunto SPS si el perfil indicado en seq_parameter_set_data() es relevante para 3DV. De acuerdo con esta divulgación, se consideran dos posibles perfiles, el "Perfil 3DV" y el "Perfil 3DV Asimétrico", para dos casos diferentes. En otras palabras, el perfil 3DV se aplica al caso de 2 visualizaciones y el perfil 3DV asimétrico se aplica al caso de 3 visualizaciones.

45 [0118] En MVC, un nuevo conjunto de parámetros de nivel de secuencia, es decir, la extensión SPS MVC se puede introducir y señalar en el subconjunto SPS. Dado que MVC se considera como una especificación base, en cualquiera de los perfiles recién agregados, el subconjunto SPS se amplía aún más para señalar una extensión 3DVC de conjunto de parámetros de secuencia, en la parte superior de la extensión SPS MVC.

50 [0119] En un códec propuesto, la nueva extensión SPS, concretamente la extensión 3DVC de conjunto de parámetros de secuencia, contiene sintaxis para reforzar las dependencias entre visualizaciones para las

conjuntos de parámetros, como SPS, conjunto de parámetros de imagen (PPS) o incluso un nuevo tipo de conjunto de parámetros, concretamente el conjunto de parámetros de visualización (VPS) si dicha información puede variar trama por trama. Si no son necesarios para la decodificación de cualquier textura o profundidad transmitida, podrían señalarse en mensajes SEI (nivel de secuencia o nivel de imagen).

5 **[0124]** Esta sección da cuenta de cómo se puede señalar la información anterior en un flujo de bits. La señalización de los parámetros de la cámara y los rangos de profundidad puede implementarse en el software pero no está habilitada para la generación de los flujos de bits.

10 **[0125]** La Tabla 4 muestra ejemplos de parámetros de cámara y rangos de profundidad en la extensión SPS 3DVC

Tabla 4

seq_parameter_set_3dvc_extension () {	C	Descriptor
[4] ...		
cam_parameters ()		
depth_ranges ()		
}		

15 **[0126]** En la tabla de sintaxis de los parámetros de la cámara, un valor de punto flotante V puede representarse con su precisión P, que es el número de dígitos antes o después del punto decimal, y un valor entero I, tal que: $V=I*10^P$. El signo de V puede ser el mismo que el de I. Dicha representación propuesta puede ser lo suficientemente precisa para los parámetros de la cámara y los rangos de profundidad y puede ser relativamente fácil de analizar y construir un valor de coma flotante.

20 **[0127]** Dado el requisito de que "los datos de vídeo fuente deben rectificarse para evitar la desalineación de la geometría y colores de la cámara " como se indica en el CfP, en esta divulgación, se puede suponer que múltiples visualizaciones comparten los mismos parámetros intrínsecos y la mayoría de los parámetros extrínsecos excepto la traslación horizontal.

25 **[0128]** La Tabla 5 y los párrafos que siguen muestran sintaxis y semántica de parámetros de cámara a modo de ejemplo.

Tabla 5

30

cam_parameters() {	C	Descritor
cam_param_present_flag	0	u(l)
if (cam_param_present_flag) {		
//parámetros intrínsecos		
focal_length_precision	0	se(v)
focal_length_x_l	0	ue(v)
focal_length_l_diffe_x	0	se(v)
principal_precision	0	se(v)
principal_poin_x_l	0	se(v)
principal_poin_y_l_diff_x	0	se(v)
//parámetros extrínsecos		
rotation_xy_half_pi	0	u(1)
rotation_xz_half_pi	0	u(1)
rotation_yz_half_pi	0	u(1)
translation-precision	0	se(v)
anchor_view_id	0	ue(v)
zero_translation-present_flag	0	u(1)

cam_parameters() {	C	Descpritor
if(!zero_translation-present_flag		
translation_anchor_view_l	0	se(v)
for(i =0; i<numViewsMinus1;i++		
if (view_id[i]!=anchor_view_id		
translation_diff_anchor_view_l[i]	0	se(v)
}		
}		

[0129] En la Tabla 5, **cam_param_present_flag** igual a 1 puede indicar que los parámetros de la cámara están señalados en este SPS. cam_param_present_flag igual a 0 puede indicar que los parámetros de la cámara no están señalados en este SPS.

[0130] En la Tabla 5, **focal_length_precision** especifica la precisión de los valores de focal_length_x y focal_length_y, que son la distancia focal de coordenadas x y la focal de coordenadas y de todas las cámaras.

[0131] En la Tabla 5, **focal_length_x_l** especifica la parte entera del valor de focal_length_x.

$$focal_length_x = focal_length_x_l * 10^{focal_length_precision}$$

[0132] En la Tabla 5, **focal_length_y_l_diff_x** más focal_length_x_l especifica la parte entera del valor de focal_length_y.

$$focal_length_y = (focal_length_x_l + focal_length_y_l_diff_x) * 10^{focal_length_precision}$$

[0133] En la Tabla 5, **principal_precision** especifica la precisión de los valores de principal_point_x y principal_point_y, que son el punto principal de la coordenada xy el punto principal de la coordenada y de todas las cámaras.

[0134] En la Tabla 5, **principal_point_x_l** especifica la parte entera del valor de principal_point_x.

$$principal_point_x = principal_point_x_l * 10^{principal_precision}$$

[0135] En la Tabla 5, **principal_point_y_l_diff_x** plus principal_point_x especifica la parte entera del valor de principal_point_y.

$$principal_point_y =$$

$$(principal_point_x_l + principal_point_y_l_diff_x) * 10^{principal_precision}$$

[0136] La matriz de rotación R para cada cámara se puede representar de la siguiente manera:

$$R = \begin{bmatrix} R_{yz} & 0 & 0 \\ 0 & R_{xz} & 0 \\ 0 & 0 & R_{xy} \end{bmatrix}$$

[0137] En la Tabla 5, **rotation_kl_half_pi** indica los elementos diagonales de la matriz de rotación **R**, con kl igual a xy, yz, o xz, en el que $R_{kl} = (-1)^{rotation_kl_half_pi}$. Este indicador igual a 0 indica $R_{kl} = 1$; este indicador igual a 1 indica $R_{kl} = -1$.

[0138] En la Tabla 5, **translation_precision** especifica la precisión de los valores de las traslaciones de todas las visualizaciones. La precisión de los valores de traslación se aplica a todos los valores de traslación de las visualizaciones que hacen referencia a este SPS.

[0139] En la Tabla 5, numViewsMinus1 se obtiene como num_views_minus1 + num_add_views_minus1 + 1.

[0140] En la Tabla 5, **anchor_view_id** especifica el view_id de la visualización, cuya traslación se usa como anclaje para calcular la traslación de las otras visualizaciones.

[0141] En la Tabla 5, **zero_translation_present_flag** igual a 1 indica que la traslación de la visualización con `view_id` igual a `anchor_view_id` es 0; este valor igual a 0 indica que se señala la traslación de la visualización con `view_id` igual a `anchor_view_id`.

5 [0142] En la Tabla 5, **translation_anchor_view_l** especifica la parte entera de la traslación de la visualización de anclaje. Denote la traslación de la visualización de anclaje como `translation_anchor_view`. `translation_anchor_view` es igual a 0 cuando `zero_translation_present_flag` es igual a 0; de lo contrario, la traslación se calcula de la siguiente manera.

10 [0143] En la Tabla 5,

$$\text{translation_anchor_view} = \text{translation_anchor_view_l} * 10^{\text{translation_precision}}$$

15 [0144] En la Tabla 5, **translation_diff_anchor_view_l[i]** más `translation_anchor_view_l` especifica la parte entera de la traslación de la visualización con `view_id` igual a `view_id[i]`, denotada como `translation_view_l[i]`.

[0145] Denote la traslación de la visualización con `view_id` igual a `view_id[i]` como `translation_view[i]`.

20 `translation_view[i] =`

$$(\text{translation_diff_anchor_view_l}[i] + \text{translation_anchor_view_l}) * 10^{\text{translation_precision}}$$

[0146] La Tabla 6 y los párrafos siguientes muestran una sintaxis y semántica de rangos de profundidad a modo de ejemplo

25

Tabla 6

<code>depth_ranges() {</code>	C	Descriptor
<code> depth_range_present_flag</code>	1	u(1)
<code> if (depth_range_present_flag) {</code>		
<code> //rango de profundidad</code>		
<code> z_near_precision</code>	1	se(v)
<code> z_far_precision</code>	1	se(v)
<code> different_depth_range_flag</code>	1	u(1)
<code> anchor_view_id</code>	1	ue(v)
<code> z_near_integer</code>	1	se(v)
<code> z_far_integer</code>	1	se(v)
<code> if (different_depth_range_flag)</code>		se(v)
<code> for(i=0; i <= numViewsMinus1 ; i++)</code>		se(v)
<code> if (view_id[i] != anchor_view_id) {</code>		
<code> z_near_diff_anchor_view_l[i]</code>	1	
<code> diff_anchor_view_l[i].</code>	1	
<code> }</code>		
<code> }</code>		
<code>}</code>		

30 [0147] En la Tabla 6, **depth_range_present_flag** igual a 1 indica que los rangos de profundidad para todas las visualizaciones se señalan en este SPS, `depth_range_present_flag` igual a 0 indica que los rangos de profundidad no se señalan en este SPS.

[0148] En la Tabla 6, **z_near_precision** especifica la precisión de un valor de `z_near`. La precisión de `z_near` como se especifica en este SPS se aplica a todos los valores `z_near` de las visualizaciones que se refieren a este SPS.

35

[0149] En la Tabla 6, **z_far_precision** especifica la precisión de un valor de `z_far`. La precisión de `z_far` como se especifica en este SPS se aplica a todos los valores `z_far` de las visualizaciones que hacen referencia a este SPS.

[0150] En la Tabla 6, **different_depth_range_flag** igual a 0 indica que los rangos de profundidad de todas las visualizaciones son los mismos y están en el rango de z_near y z_far , inclusive. **different_depth_range_flag** igual a 1 indica que los rangos de profundidad de todas las visualizaciones pueden ser diferentes: z_near y z_far son el rango de profundidad para la visualización de anclaje, y $z_near[i]$ y $z_far[i]$ se especifican adicionalmente en este SPS como el rango de profundidad de una visualización con $view_id$ igual a $view_id[i]$.

[0151] En la Tabla 6, **z_near_integer** especifica la parte entera del valor de z_near . $z_near = z_near_integer * 10^{z_near_precision}$.

[0152] En la Tabla 6, **z_far_integer** especifica la parte entera del valor de z_far . $z_far = z_far_integer * 10^{z_far_precision}$.

[0153] En la Tabla 6, **z_near_diff_anchor_view_l** más $z_near_integer$ especifica la parte entera del valor de profundidad más cercano de la visualización con $view_id$ igual a $view_id[i]$, denotado como $z_near_l[i]$.

[0154] Denota el z_near de la visualización con $view_id$ igual a $view_id[i]$ ya que $z_near[i]$. $z_near[i] = (z_near_diff_anchor_view_l[i] + z_near_integer) * 10^{z_near_precision}$.

[0155] En la Tabla 6, **z_far_diff_anchor_view_l** plus $z_far_integer$ especifica la parte entera del valor de profundidad más lejano de la visualización con $view_id$ igual a $view_id[i]$, denotado como $z_far_l[i]$.

$$z_far[i] = (z_far_diff_anchor_view_l[i] + z_far_integer) * 10^{z_far_precision}$$

[0156] La Tabla 7 muestra una sintaxis de RBSP de conjunto de parámetros de visualización a modo de ejemplo.

Tabla 7

view_parameter_set_rbsp() {	C	Descriptor
view_parameter_set()		
}		

A una unidad NAL que contiene esta RBSP de conjunto de parámetros de visualización se le puede asignar un nuevo tipo de unidad NAL, por ejemplo, 16.

[0157] La Tabla 8 y los párrafos siguientes muestran la sintaxis y semántica de conjunto de parámetros de visualización a modo de ejemplo.

Tabla 8

view_parameter_set(){	C	Descriptor
seq_para_set_id	1	ue(v)
for(i=0; i<=numViewsMinus1;i++ {		
delta_translation_update_view_l[i]	1	se(v)
z_near_update_view_l[i]	1	se(v)
z_far_update_vie_x_l[i]	1	se(v)
}		
rbsp_trailing_bits()	1	
}		

[0158] El rango de profundidad y la traslación de una cámara pueden cambiar en una imagen. El rango de profundidad o los parámetros de la cámara pueden actualizados ser aplicables a los componentes de visualización de la unidad de acceso actual y los siguientes componentes de visualización en el flujo de bits, hasta que un nuevo VPS después del VPS actual actualice esos valores para las visualizaciones relacionadas.

[0159] Por simplicidad, la semántica de los elementos sintácticos no está dada. Para la traslación o el rango de profundidad de cada visualización, la parte entera de la diferencia entre el nuevo valor y el valor señalado en el SPS (con un identificador igual a $seq_para_set_id$), se puede señalar en este VPS. Los valores actualizados de la traslación y el rango de profundidad se pueden calcular de la siguiente manera:

$$translation_view[i] =$$

$$(translation_view_integer[i] + translation_update_view_l[i]) * 10^{translation_precision}$$

$$z_near[i]=$$

$$(z_near_integer[i] + z_near_update_view_l[i]) * 10^{z_far_precision}$$

$$z_far[i] = (z_far_integer[i] + z_far_update_view_l[i]) * 10^{z_far_precision}$$

10 en el que $translation_view_integer[i]$, $z_near_integer[i]$ y $z_far_integer[i]$ son las partes enteras de los valores de $translation_view[i]$, $z_near[i]$, y $z_far[i]$, que se calculan basándose en la señalización en SPS.

15 **[0160]** Una o más de las técnicas de esta divulgación se pueden usar para proporcionar mejoras de codificación en términos de compresión y/o calidad. El tiempo de codificación y la complejidad también se pueden mejorar usando una o más de las técnicas de esta divulgación. El tiempo de descodificación y la complejidad también pueden mejorarse. Además, la cantidad de uso de memoria en el codificador y el descodificador se puede mejorar o reducir en relación con otras técnicas.

20 **[0161]** En algunos ejemplos, tanto el codificador como el descodificador pueden tener el mismo nivel de consumo de memoria que el codificador y descodificador JMVC. Por lo tanto, el uso de memoria se puede considerar como proporcional al número de componentes de visualización, por ejemplo, en una unidad de acceso. Si el componente de visualización de profundidad siempre se almacena como 4:0:0, con el mismo número de visualizaciones, las soluciones propuestas pueden consumir aproximadamente 5/3 (aproximadamente 67 % de aumento) de la memoria utilizada por JMVC para codificador o descodificador. Tenga en cuenta que para la simplicidad de las operaciones, por ejemplo, visualizar los mapas de profundidad y usarlos para visualizar la síntesis, el codificador y el descodificador aún pueden tomar y retirar los archivos de profundidad en un formato de muestreo de crominancia 4:2:0.

30 **[0162]** Ahora se describirán las características de complejidad del descodificador. En algunos ejemplos, tanto el codificador como el descodificador coherentes con las técnicas de esta divulgación pueden tener el mismo nivel de complejidad que el codificador y descodificador JMVC. Cuando se compara con JMVC, la complejidad de cálculo de un códec coherente con esta divulgación puede estar relacionada con el número de visualizaciones y la resolución espacial de cada visualización. En otras palabras, un códec coherente con esta divulgación puede requerir la misma cantidad de cálculos que el códec JMVC, siempre que ambos tomen el mismo vídeo con el mismo número de píxeles.

35 **[0163]** En el lado del descodificador, puede ser necesario un muestreo ascendente de nivel de imagen normativo para el perfil 3DV asimétrico. Sin embargo, tal proceso de descodificación podría considerarse menos complejo que los otros procesos de descodificación para la descodificación de un componente de visualización de alta resolución, por lo que las características de complejidad pueden todavía indicarse, por ejemplo, mediante cuántos MB se procesarán por segundo.

40 **[0164]** Un codificador consistente con las técnicas descritas en el presente documento puede seguir el esquema de codificador JMVC actual, en el que las visualizaciones se codifican una a una. Dentro de cada visualización, la secuencia de textura se codifica primero, y la secuencia de profundidad se codifica a continuación.

50 **[0165]** Cuando el modo IVMP está habilitado, durante la codificación del componente de visualización de textura, el movimiento archivado de cada componente de visualización de textura se escribe en un archivo de movimiento, cuyo nombre se puede especificar en el archivo de configuración. Al codificar la secuencia de profundidad asociada de la misma visualización, el archivo de movimiento se lee para referencia.

[0166] El codificador puede usar la misma configuración que JMVC con los siguientes elementos adicionales.

MotionFile

Secuencia, por defecto: "movimiento"

60 **[0167]** Especifica el nombre del archivo (sin el.dat) de la secuencia de movimiento que se generará. Esta secuencia se proporciona para el modo IVMP. *motion_0.dat*, *motion_1.dat*, etc. serán creados por el codificador automáticamente.

HalfSizeDimension

Unsigned Int, por defecto: 0

65

[0168] Indica si se usa resolución espacial asimétrica y si se usa, la dimensión de submuestreo. Los siguientes valores tienen soporte:

- 0 - Todas las visualizaciones están codificadas con la misma resolución espacial.
- 1 - Se usa una resolución espacial asimétrica y las visualizaciones de resolución media tienen una anchura media de las otras visualizaciones.
- 2 - Se usa una resolución espacial asimétrica y las visualizaciones de resolución media tienen una altura media de las otras visualizaciones.

10 BasisQP_texture

Doble, por defecto: 26

[0169] Especifica el parámetro de cuantificación básico del componente de visualización de textura con resolución espacial media.

15 BasisQP_depth

Doble, por defecto: 26

[0170] Especifica el parámetro de cuantificación básico del componente de visualización de profundidad con resolución espacial media.

20 BasisQP_texture_delta

Unsigned Int, por defecto: 0

[0171] Especifica la desviación del parámetro de cuantificación básico para el parámetro de cuantificación básico del componente de visualización de textura con resolución espacial completa, en comparación con el parámetro de cuantificación básico del componente de visualización de textura con resolución espacial media. El parámetro de cuantificación básico del componente de visualización de textura con resolución espacial completa se calcula mediante $BasisQP_texture(resolución\ espacial\ completa) = BasisQP_texture + BasisQP_texture_delta$.

25 BasisQP_depth_delta

Sin signo, por defecto: 0

[0172] Especifica la desviación del parámetro de cuantificación básico para el parámetro de cuantificación básico del componente de visualización de profundidad con resolución espacial completa, en comparación con el parámetro de cuantificación básico del componente de visualización de profundidad con resolución espacial media. El parámetro de cuantificación básico del componente de visualización de profundidad con resolución espacial completa se calcula mediante $BasisQP_depth(full\ spatial\ resolution) = BasisQP_depth + BasisQP_depth_delta$.

30 NoDepthInterViewFlag

Indicador (0 o 1), por defecto: 0

[0173] Especifica si la predicción entre visualizaciones está habilitada para cualquier componente de visualización de profundidad. Cuando *NoDepthInterViewFlag* es igual a 0, la predicción entre visualizaciones está habilitada. Cuando *NoDepthInterViewFlag* es igual a 1, la predicción entre visualizaciones está deshabilitada.

35 HalfRes

Indicador (0 o 1), por defecto: 0

[0174] Este valor está asociado con un valor de *View_ID* como parte de la propiedad de cada visualización de referencia señalada en la porción de dependencia de visualización.

[0175] Especifica si la visualización identificada por *View_ID* es una resolución *espacial* media. Cuando *HalfRes* es igual a 0, es una visualización de resolución espacial completa. Cuando *HalfRes* es igual a 1, es una visualización de resolución espacial media.

[0176] El codificador se puede usar para generar los flujos de bits. Una llamada de codificador a modo de ejemplo se ilustra en el siguiente ejemplo.

H264AVCEncoderLibTestStatic.exe -vf <mcfg> <view_id> <component_idx>

[0177] Aquí, *mcfg* representa el nombre de archivo del archivo de configuraciones. El archivo de configuración se puede especificar para cada llamada del codificador. El elemento *view_id* representa la visualización que se codificará. El elemento *component_idx* indica si la secuencia actual a codificar es textura (cuando *component_idx* es igual a 1) o profundidad (cuando *component_idx* es igual a 0) de la visualización específica. El codificador se puede ejecutar para cada componente de visualización de cada visualización que se va a codificar.

[0178] El descodificador puede ser similar al descodificador JMVC, con la modificación principal de descodificar y emitir también una secuencia de profundidad para cada visualización. En el perfil asimétrico 3DV, se necesita un muestreo ascendente para convertir la visualización MVC (izquierda o derecha) a una resolución alta para la predicción de la visualización adicional (centro).

[0179] Un ensamblador puede tener modificaciones muy pequeñas para descartar las unidades NAL del conjunto de parámetros duplicados; la complejidad es la misma que para el ensamblador JMVC.

[0180] Con respecto a una visualización, las modificaciones del sintetizador relativas a JMVC pueden no ser necesarias.

[0181] Se han descrito varias características para un códec 3DVC basado en H.264/MVC, que pueden cumplir con todos los requisitos de "deber" de la propuesta, y pueden proporcionar un buen rendimiento de codificación con una cantidad relativamente pequeña de procedimientos de codificación adicionales. Los procedimientos incluyen el marco de alto nivel para la codificación conjunta de textura y profundidad, la predicción de textura a profundidad dentro de un componente de visualización y la predicción entre visualizaciones entre componentes de visualización de textura o profundidad con resoluciones espaciales asimétricas.

[0182] Un códec 3DV basado en MVC puede estandarizarse para las necesidades del mercado a corto plazo y las características propuestas de esta divulgación pueden ser la base del software de referencia y del borrador de trabajo de dicho códec 3DV.

[0183] La FIG. 9 es un diagrama de flujo que ilustra una técnica que puede ser realizada por un codificador de vídeo consistente con esta divulgación. La FIG. 9 se describirá desde la perspectiva del codificador de vídeo 20 de la FIG. 2, aunque también podrían usarse otros codificadores de vídeo. Como se muestra en la FIG. 9, el módulo de predicción 41 recibe vídeo 3D, por ejemplo, bloques de vídeo que representan la reproducción 3D (901). El vídeo 3D incluye un bloque de vídeo de visualización de textura y un bloque de vídeo de visualización de profundidad asociado (901). El módulo de predicción 41 codifica el bloque de vídeo de visualización de textura (902). Además, el módulo de predicción 41 codifica el bloque de vídeo de visualización de profundidad (903).

[0184] De acuerdo con esta divulgación, el módulo de predicción 41 soporta un modo IVMP. En particular, el módulo de predicción 41 genera un elemento sintáctico para indicar si se adopta información de movimiento para la visualización de profundidad desde la visualización de textura (903). De esta forma, si el modo IMVP está habilitado, el componente de visualización de profundidad puede no incluir ningún valor delta adicional con respecto a su información de movimiento, y en su lugar, puede adoptar la información de movimiento del componente de visualización de textura como su información de movimiento. En particular, con el modo IMVP, el componente de visualización de profundidad puede no incluir ningún valor diferente de vector de movimiento, pero puede adoptar completamente el vector de movimiento del componente de visualización de textura correspondiente. Al definir un modo que adopta completamente la información de movimiento de una visualización de textura como la información de movimiento de una visualización de profundidad, sin ninguna señalización de los valores delta del vector de movimiento con respecto a dicha información de movimiento, puede lograrse una compresión mejorada.

[0185] El bloque de vídeo de visualización de textura y el bloque de vídeo de visualización de profundidad pueden codificarse juntos en la unidad de capa de abstracción de red (NAL), y el elemento sintáctico puede comprender un indicador en la unidad NAL para indicar si la información de movimiento asociada con el bloque de vídeo de visualización de textura se adopta como la información de movimiento asociada con el bloque de vídeo de visualización de profundidad. En este caso, si el elemento sintáctico indica que la información de movimiento asociada con el bloque de vídeo de visualización de textura se adopta como la información de movimiento asociada con el bloque de vídeo de visualización de profundidad, el bloque de vídeo de visualización de profundidad no incluye ningún delta adicional con respecto a la información de movimiento asociada con el bloque de vídeo de visualización de profundidad. Las unidades NAL son un tipo específico de unidad de acceso utilizada para codificar datos de vídeo, y las técnicas también se pueden usar con otros tipos de unidades de vídeo.

[0186] Más específicamente, el elemento sintáctico puede comprender uno o más bits que indiquen si el modo de IVMP está habilitado. Si el modo IVMP está deshabilitado, la información de movimiento asociada con el bloque de vídeo de visualización de textura se incluye en la unidad NAL y la información de movimiento asociada con el bloque de vídeo de visualización de profundidad se incluye por separado en la unidad NAL. De forma alternativa, si el modo IVMP está habilitado, la información de movimiento asociada con el bloque de vídeo de visualización de textura se incluye en la unidad NAL y la información de movimiento asociada con el bloque de vídeo de visualización de textura

se adopta como la información de movimiento asociada con el bloque de vídeo de visualización de profundidad. En consecuencia, si el modo IVMP está habilitado, el bloque de vídeo de visualización de profundidad no incluye ningún delta adicional con respecto a la información de movimiento asociada con el bloque de vídeo de visualización de profundidad. En algunos ejemplos, el modo IVMP solo se aplica a las imágenes que no son de anclaje y no se aplica a las imágenes de anclaje.

[0187] La FIG. 10 es un diagrama de flujo que ilustra una técnica que puede ser realizada por un descodificador de vídeo consistente con esta divulgación. La FIG. 10 se describirá desde la perspectiva del descodificador de vídeo 30 de la FIG. 3, aunque también podrían usarse otros descodificadores de vídeo. Como se muestra en la FIG. 10, el módulo de predicción 81 recibe vídeo 3D, por ejemplo, bloques de vídeo que representan datos de vídeo 3D (1001). El vídeo 3D incluye un bloque de vídeo de visualización de textura y un bloque de vídeo de visualización de profundidad asociado (1001). El módulo de predicción 41 descodifica el bloque de vídeo de visualización de textura (1002). Además, el módulo de predicción 41 descodifica el bloque de vídeo de visualización de profundidad (1003).

[0188] De acuerdo con esta divulgación, el módulo de predicción 81 soporta un modo IVMP. En particular, el módulo de predicción 81 descodifica un elemento sintáctico para indicar si se adopta información de movimiento para la visualización de profundidad desde la visualización de textura (1003). El descodificador puede interpretar el elemento sintáctico como indicando si la información de movimiento de la visualización de profundidad se adopta desde la visualización de textura. Si el modo IMVP está habilitado, el componente de visualización de profundidad puede no incluir ningún valor delta adicional con respecto a su información de movimiento, y en su lugar, puede adoptar la información de movimiento del componente de visualización de textura como su información de movimiento. De nuevo, definiendo un modo que adopta completamente información de movimiento de una visualización de textura como la información de movimiento de una visualización de profundidad, sin ninguna señalización de valores delta con respecto a dicha información de movimiento, se puede lograr una compresión mejorada.

[0189] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de estos. Si se implementan en software, las funciones pueden almacenarse en o transmitirse a través de, como una o más instrucciones o código, en un medio legible por ordenador o ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como unos medios de almacenamiento de datos o unos medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) unos medios de almacenamiento tangibles legibles por ordenador que son no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser medios disponibles cualesquiera a los que se puede acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0190] A modo de ejemplo, y no de manera limitativa, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. También, cualquier conexión recibe debidamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otro origen remoto usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. El término disco, tal como se utiliza en el presente documento, incluye un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, donde algunos discos habitualmente emiten datos magnéticamente, mientras que otros discos emiten datos ópticamente con láseres. Las combinaciones de lo anterior deberían incluirse también dentro del alcance de los medios legibles por ordenador.

[0191] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), matrices lógicas programables por campo (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para la codificación y la descodificación, o incorporarse en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

5 **[0192]** Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por medio de un grupo de unidades de hardware interoperativas, que incluyen uno o más procesadores como los descritos anteriormente, conjuntamente con software y/o firmware adecuados.

10 **[0193]** En todavía otros ejemplos, esta divulgación puede dirigirse a un medio de almacenamiento legible por ordenador que almacena estructuras de datos en el mismo. Las estructuras de datos pueden incluir datos de vídeo 3D que se comprimen de la manera descrita en el presente documento, tal como mediante el uso del modo IVMP para codificar una visualización de profundidad con respecto a una visualización de textura.

15 **[0194]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento para codificar datos de vídeo tridimensionales, 3D, que comprende:
 - 5 codificar un bloque de vídeo de visualización de textura; y
 codificar un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura,
 en el que codificar el bloque de vídeo de visualización de profundidad incluye codificar un elemento
 10 sintáctico indicativo de si un modo de predicción de movimiento de visualización interna, IVMP está
 habilitado de modo que la información de movimiento asociada con el bloque de vídeo de visualización de
 textura se adopte como información de movimiento asociada con el bloque de vídeo de visualización de
 profundidad,
 en el que el bloque de vídeo de visualización de textura y el bloque de vídeo de visualización de
 profundidad están codificados conjuntamente en una unidad,
 15 en el que si el modo IVMP está deshabilitado, la información de movimiento asociada con el bloque de
 vídeo de visualización de textura se incluye en la unidad y la información de movimiento asociada con el
 bloque de vídeo de visualización de profundidad se incluye por separado en la unidad, y
 en el que si el modo IVMP está habilitado, la información de movimiento asociada con el bloque de vídeo
 de visualización de textura se incluye en la unidad, y la información de movimiento asociada con el bloque
 20 de vídeo de visualización de textura se adopta como la información de movimiento asociada con el bloque
 de vídeo de visualización de profundidad.
 2. El procedimiento según la reivindicación 1, en el que el elemento sintáctico comprende un indicador definido
 a nivel de bloque de vídeo, siendo el indicador indicativo de si el modo IVMP está habilitado, de modo que la
 25 información de movimiento asociada con el bloque de vídeo de visualización de textura se adopte como la
 información de movimiento asociada con el bloque de vídeo de visualización de profundidad.
 3. El procedimiento según la reivindicación 1, en el que la codificación comprende codificación, y en el que la
 codificación del elemento sintáctico comprende generar el elemento sintáctico.
 30
 4. El procedimiento según la reivindicación 1, en el que la codificación comprende descodificación, en el que la
 codificación del elemento sintáctico comprende descodificar el elemento sintáctico de un flujo de bits
 codificado, y en el que el elemento sintáctico está incluido en el flujo de bits codificado.
 - 35 5. Un dispositivo configurado para codificar datos de vídeo tridimensionales, 3D, comprendiendo el dispositivo:
 - medios para codificar un bloque de vídeo de visualización de textura; y
 medios para codificar un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de
 visualización de profundidad está asociado con el bloque de vídeo de visualización de textura,
 40 en el que los medios para codificar el bloque de vídeo de visualización de profundidad incluyen medios
 para generar un elemento sintáctico indicativo de si un modo de predicción de movimiento de
 visualización interna, IVMP está habilitado de modo que la información de movimiento asociada con el
 bloque de vídeo de visualización de textura se adopte como información de movimiento asociada con el
 bloque de vídeo de visualización de profundidad,
 45 en el que el bloque de vídeo de visualización de textura y el bloque de vídeo de visualización de
 profundidad están codificados conjuntamente en una unidad,
 en el que si el modo IVMP está deshabilitado, la información de movimiento asociada con el bloque de
 vídeo de visualización de textura se incluye en la unidad y la información de movimiento asociada con el
 bloque de vídeo de visualización de profundidad se incluye por separado en la unidad, y
 50 en el que si el modo IVMP está habilitado, la información de movimiento asociada con el bloque de vídeo
 de visualización de textura se incluye en la unidad, y la información de movimiento asociada con el bloque
 de vídeo de visualización de textura se adopta como la información de movimiento asociada con el bloque
 de vídeo de visualización de profundidad.
 - 55 6. El dispositivo según la reivindicación 5, en el que el elemento sintáctico comprende un indicador definido en
 un nivel de bloque de vídeo, siendo el indicador indicativo de si el modo IVMP está habilitado, de modo que la
 información de movimiento asociada al bloque de vídeo de visualización de textura se adopte como la
 información de movimiento asociada con el bloque de vídeo de visualización de profundidad.
 - 60 7. El dispositivo según la reivindicación 5 o reivindicación 6, en el que los medios para codificar el bloque de
 vídeo de visualización de textura, los medios para codificar el bloque de vídeo de visualización de
 profundidad y los medios para generar el elemento sintáctico comprenden uno o más procesadores
 configurados para:
 - 65 codificar el bloque de vídeo de visualización de textura;
 codificar el bloque de vídeo de visualización de profundidad; y

generar el elemento sintáctico indicativo de si el modo IVMP está habilitado.

- 5
8. Un dispositivo configurado para descodificar datos de vídeo tridimensionales, 3D, comprendiendo el dispositivo:
- medios para descodificar un bloque de vídeo de visualización de textura; y
 medios para descodificar un bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de profundidad está asociado con el bloque de vídeo de visualización de textura, en el que los medios para descodificar el bloque de vídeo de visualización de profundidad incluyen medios para descodificar un elemento sintáctico incluido en un flujo de bits codificado, siendo el elemento sintáctico indicativo de si un modo de predicción de movimiento de visualización interna, IVMP, está habilitado de tal manera que la información de movimiento asociada con el bloque de vídeo de visualización de textura se adopta como información de movimiento asociada con el bloque de vídeo de visualización de profundidad, en el que el bloque de vídeo de visualización de textura y el bloque de vídeo de visualización de profundidad están codificados conjuntamente en una unidad, en el que si el modo IVMP está deshabilitado, la información de movimiento asociada con el bloque de vídeo de visualización de textura se incluye en la unidad y la información de movimiento asociada con el bloque de vídeo de visualización de profundidad se incluye por separado en la unidad, y en el que si el modo IVMP está habilitado, la información de movimiento asociada con el bloque de vídeo de visualización de textura se incluye en la unidad, y la información de movimiento asociada con el bloque de vídeo de visualización de profundidad se adopta como la información de movimiento asociada con el bloque de vídeo de visualización de profundidad.
- 10
- 15
- 20
- 25 9. El dispositivo según la reivindicación 8, en el que el elemento sintáctico comprende un indicador definido en un nivel de bloque de vídeo, siendo el indicador indicativo de si el modo IVMP está habilitado, de modo que la información de movimiento asociada al bloque de vídeo de visualización de textura se adopte como la información de movimiento asociada con el bloque de vídeo de visualización de profundidad.
- 30 10. El dispositivo según cualquiera de las reivindicaciones 8 y 9, en el que los medios para descodificar el bloque de vídeo de visualización de textura, los medios para descodificar el bloque de vídeo de visualización de profundidad y los medios para descodificar el elemento sintáctico comprenden uno o más procesadores configurados para:
- 35 descodificar el bloque de vídeo de visualización de textura;
 descodificar el bloque de vídeo de visualización de profundidad; y
 descodificar el elemento sintáctico indicativo de si el modo IVMP está habilitado.
- 40 11. El dispositivo según cualquiera de las reivindicaciones 5 a 10, en el que el dispositivo comprende un teléfono inalámbrico.
- 45 12. El dispositivo según cualquiera de las reivindicaciones 5 a 10, en el que el dispositivo comprende uno o más de:
- un televisor digital,
 un dispositivo en el sistema de radiodifusión directa digital,
 un dispositivo en el sistema de radiodifusión inalámbrica,
 un asistente digital personal, PDA,
 un ordenador portátil,
 un ordenador de escritorio,
 una tableta,
 un lector de libros electrónicos,
 una cámara digital,
 un dispositivo de grabación digital,
 un reproductor de medios digitales,
 un dispositivo de videojuegos,
 una consola de videojuegos,
 un teléfono de radio celular,
 un teléfono de radio satelital,
 un teléfono inteligente,
 un dispositivo de videoconferencia, y
 un dispositivo de transmisión de vídeo.
- 50
- 55
- 60
- 65 13. Un medio de almacenamiento legible por ordenador que comprende instrucciones almacenadas en el mismo, en el que las instrucciones en la ejecución hacen que uno o más procesadores realicen el procedimiento de acuerdo con una cualquiera de las reivindicaciones 1 a 4.

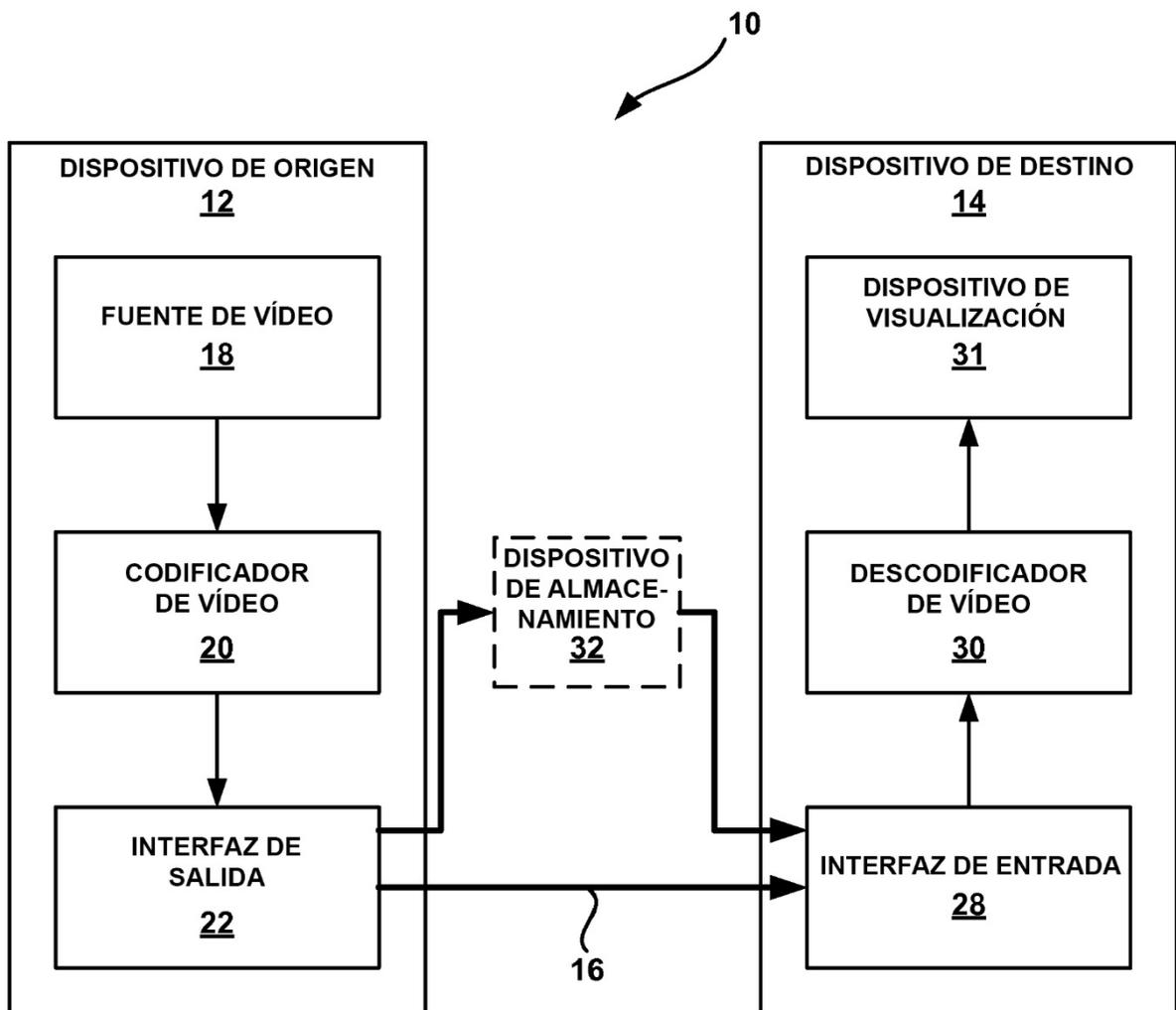


FIG. 1

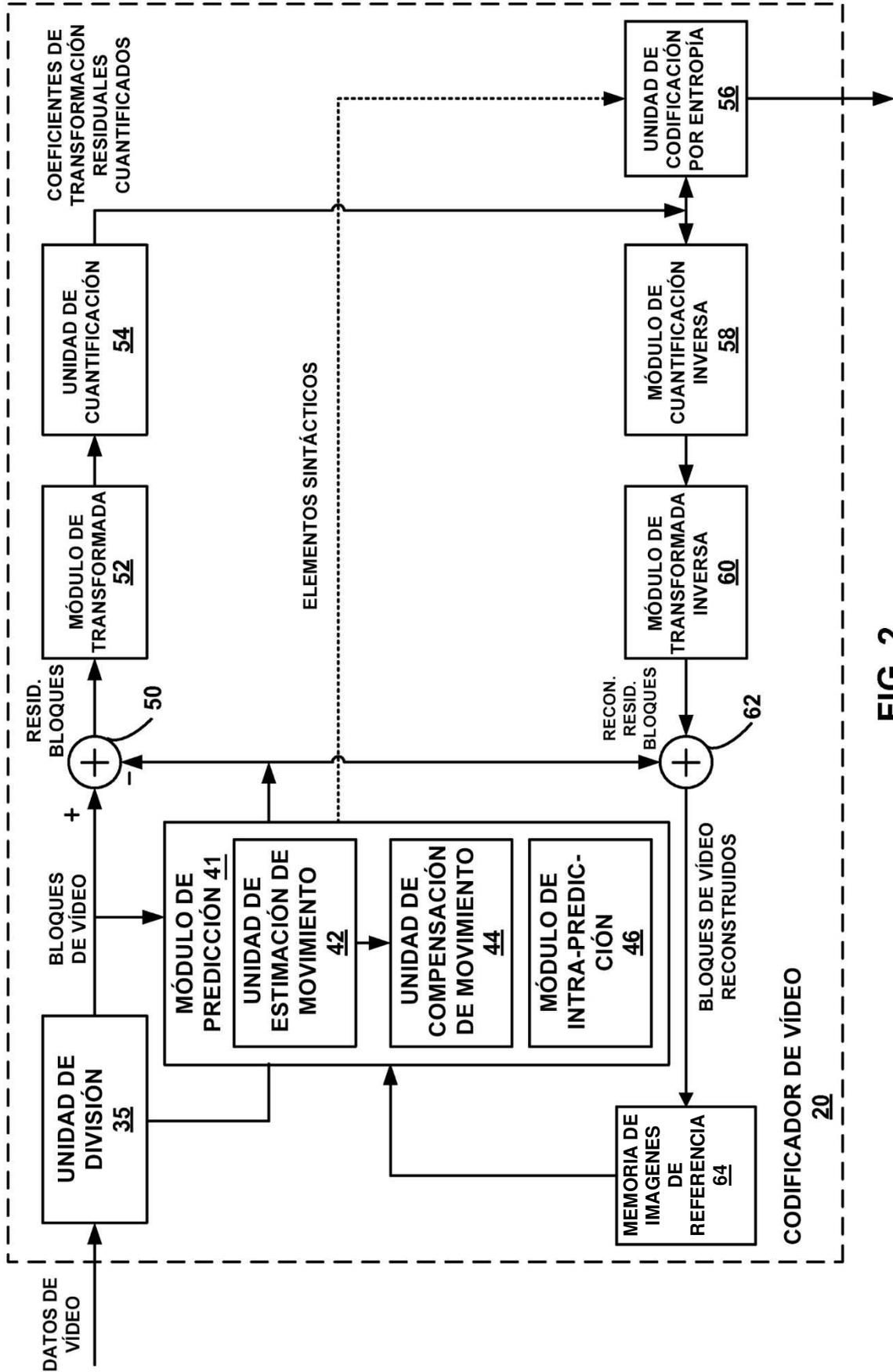


FIG. 2

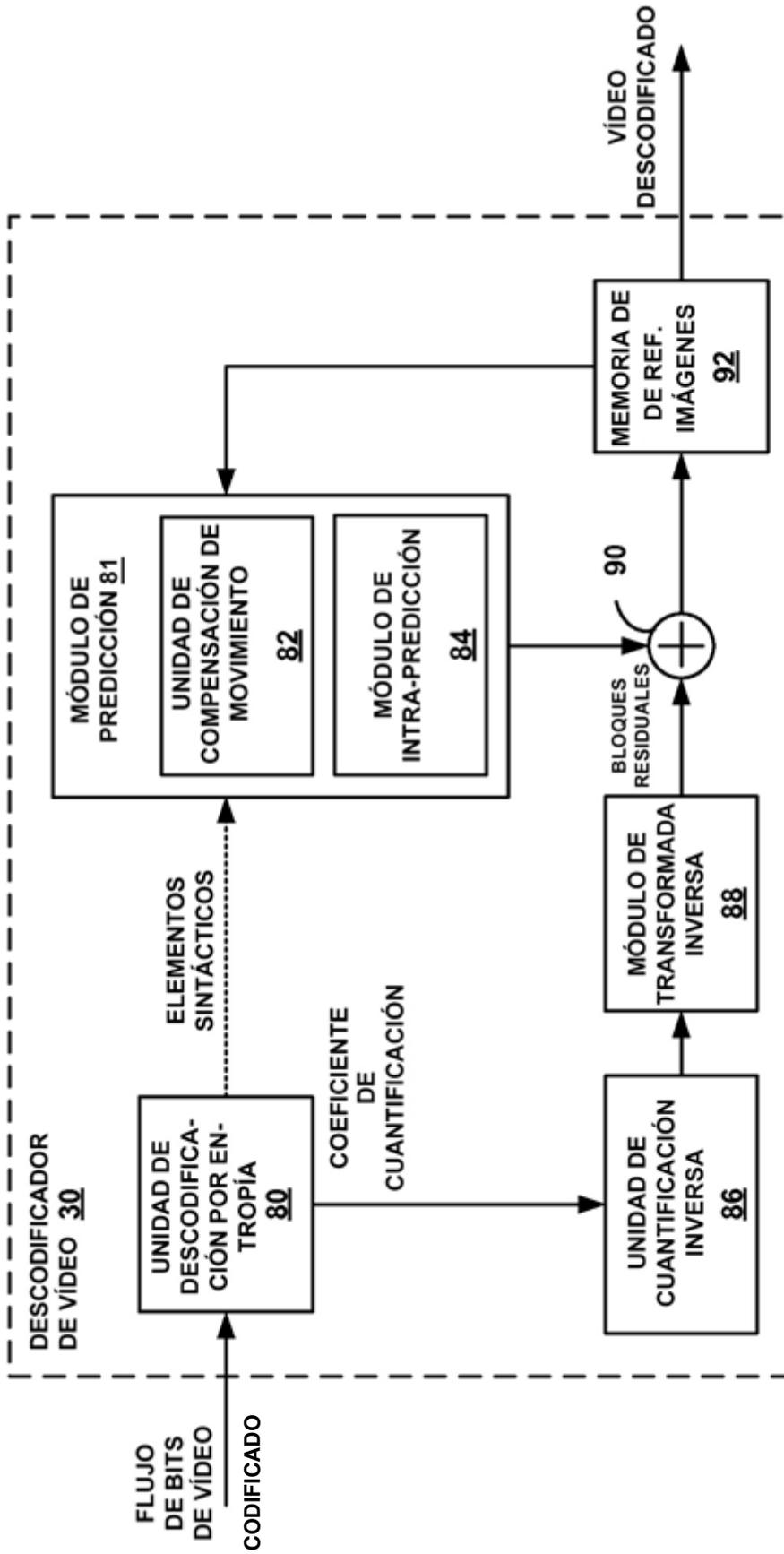


FIG. 3

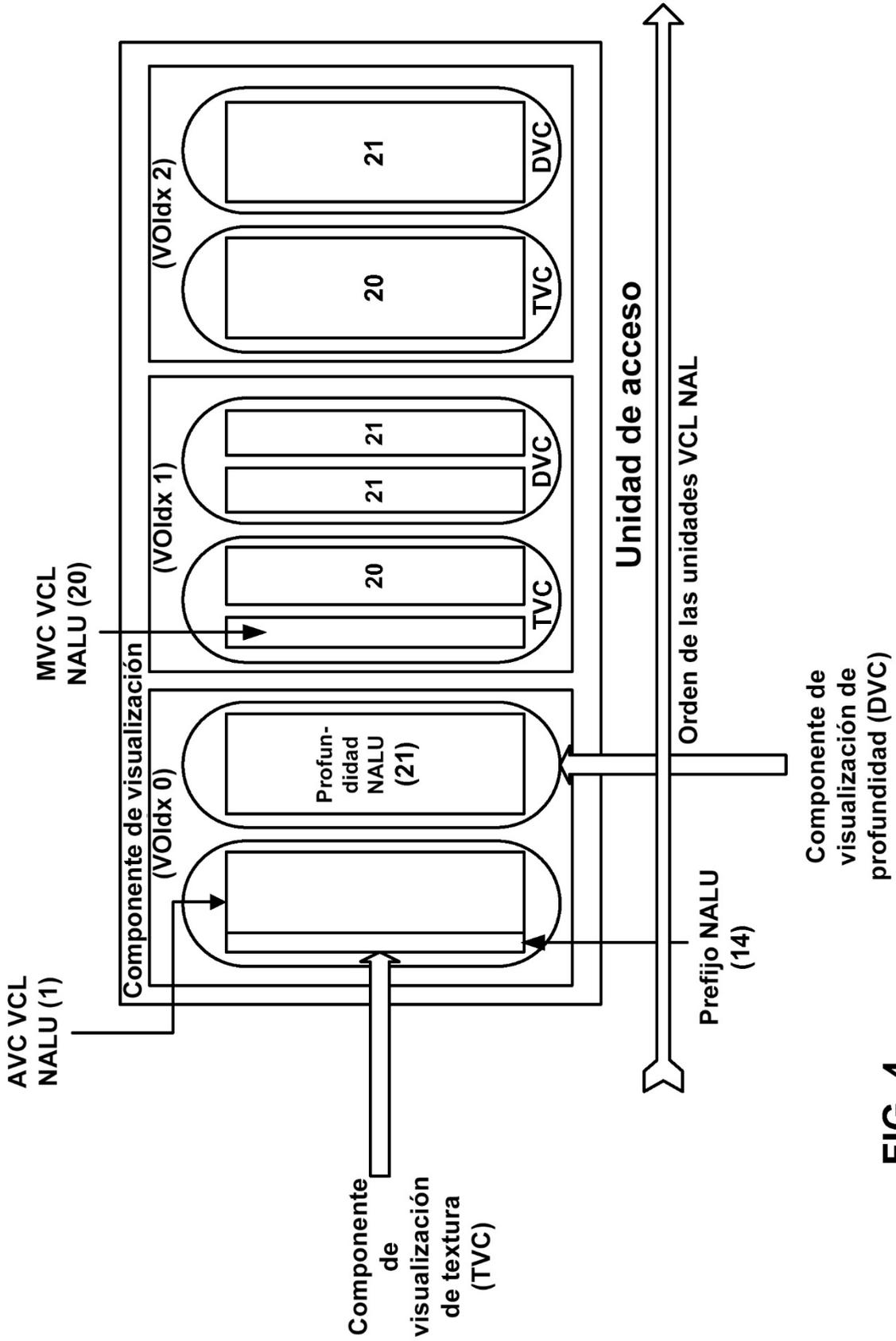


FIG. 4

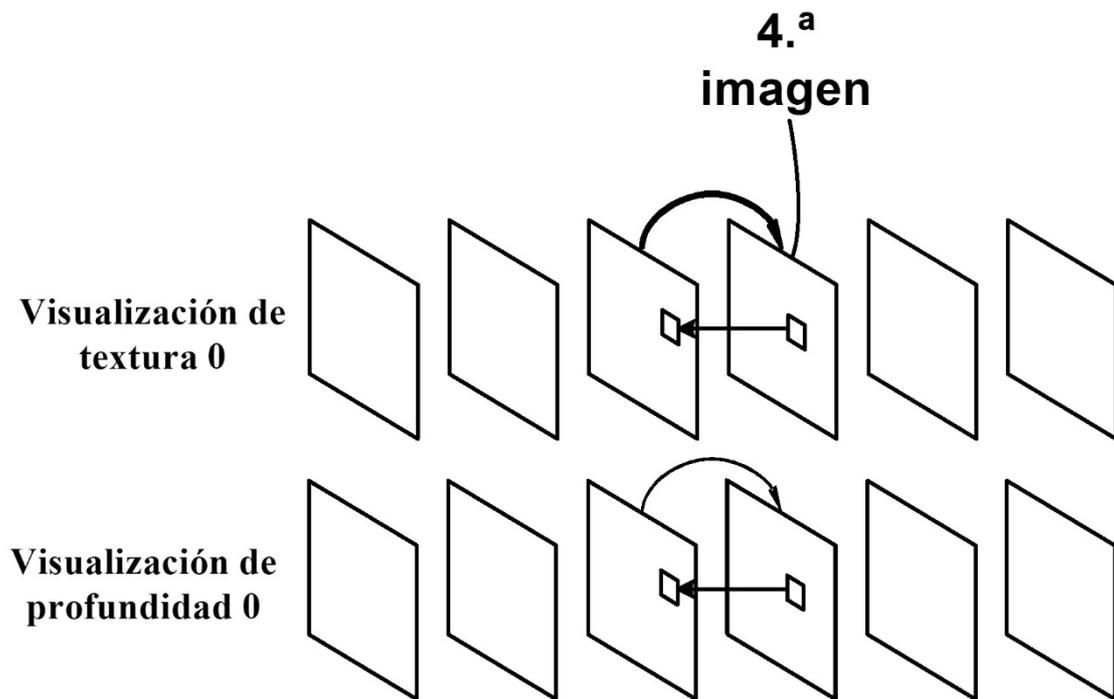


FIG. 5

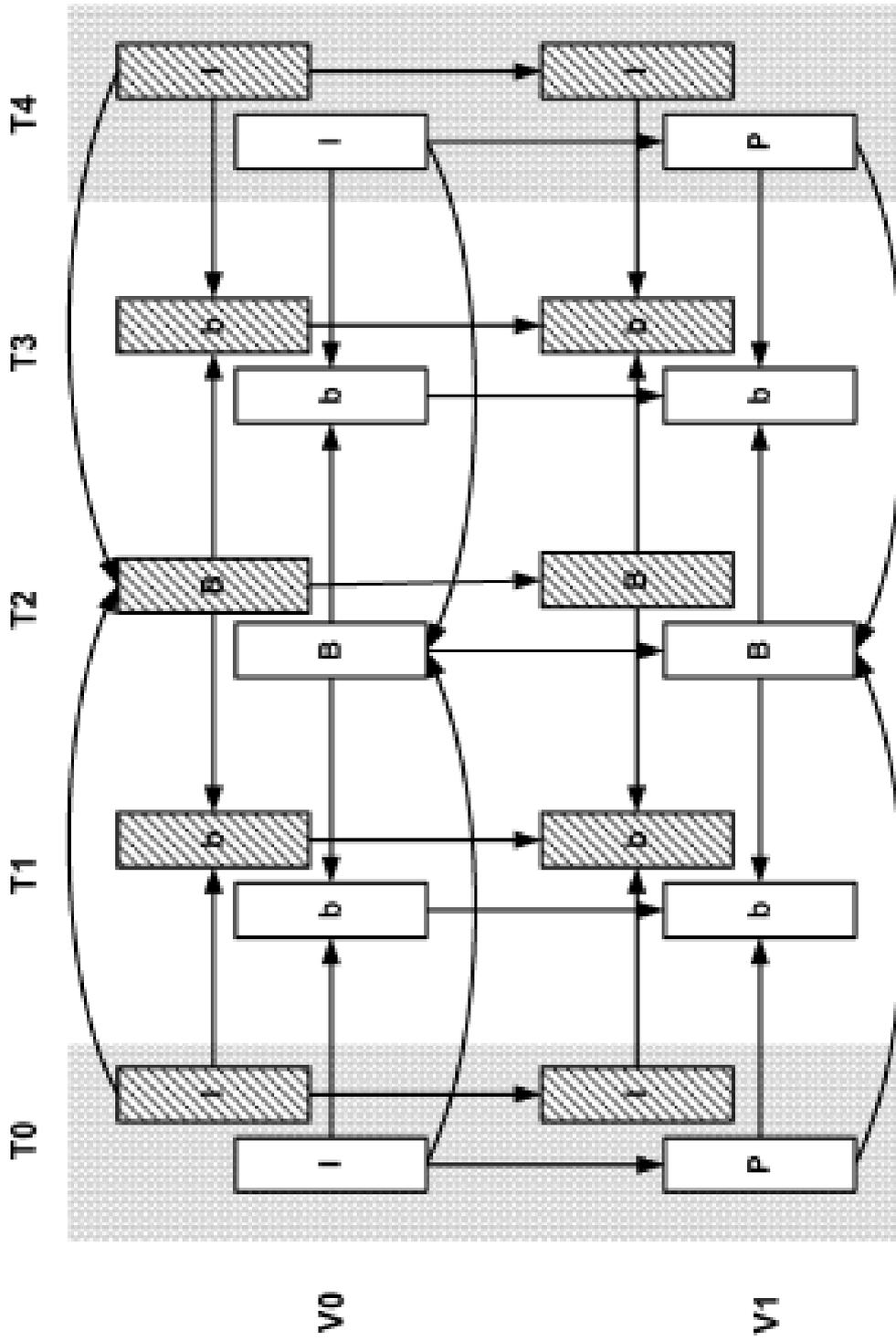


FIG. 6

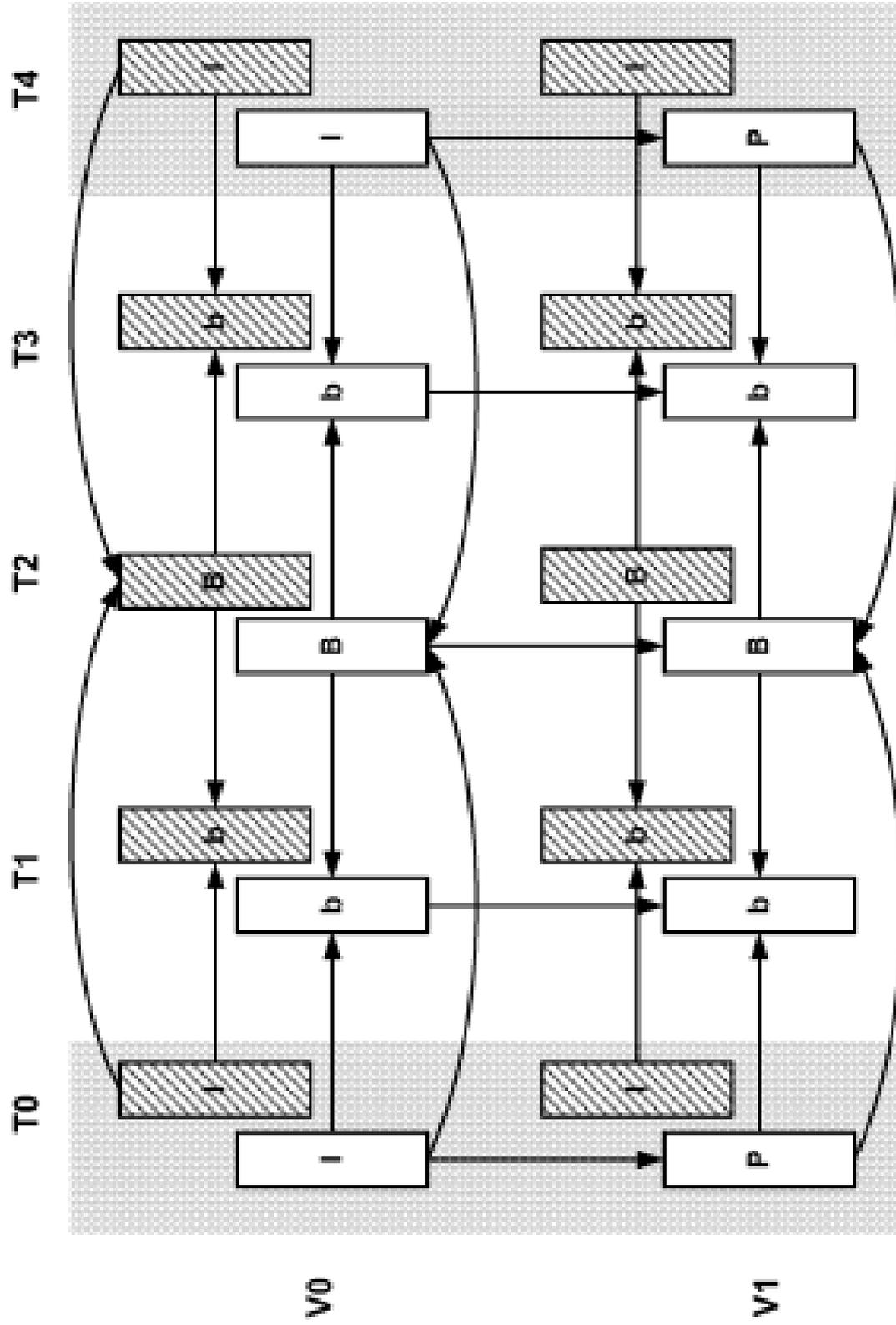


FIG. 7

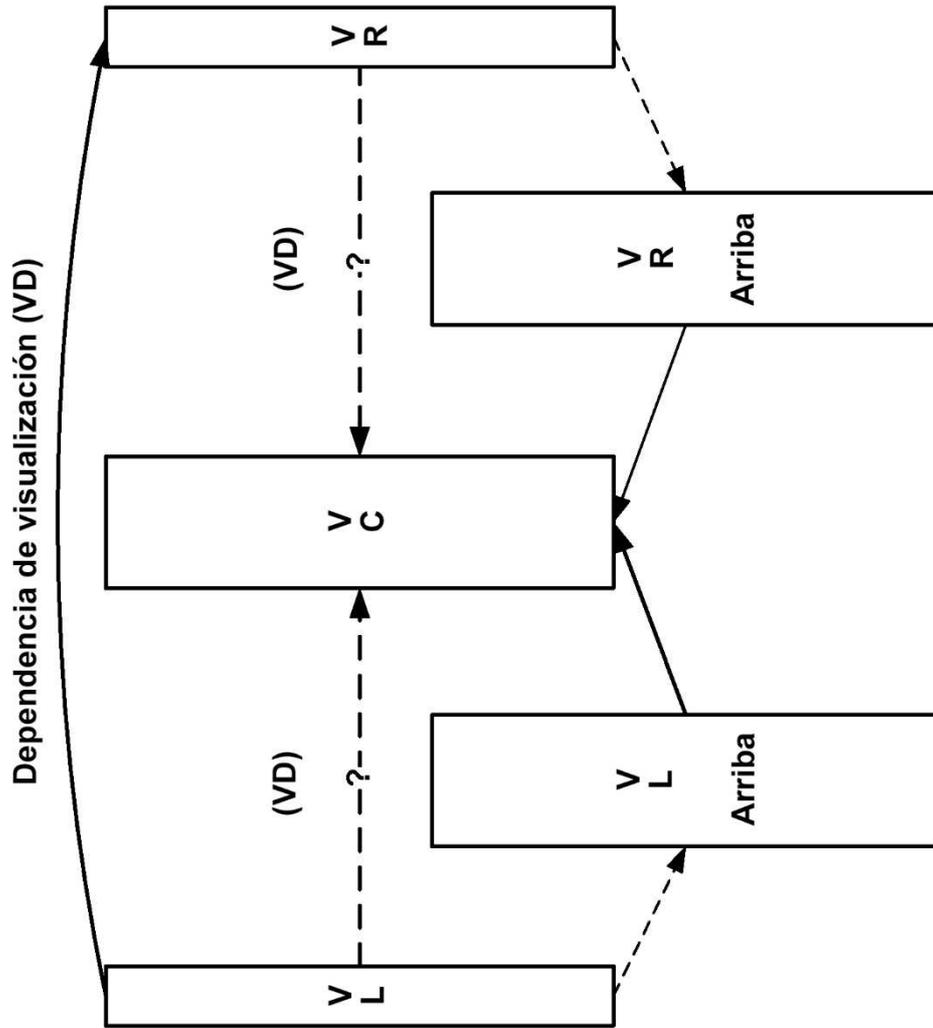


FIG. 8

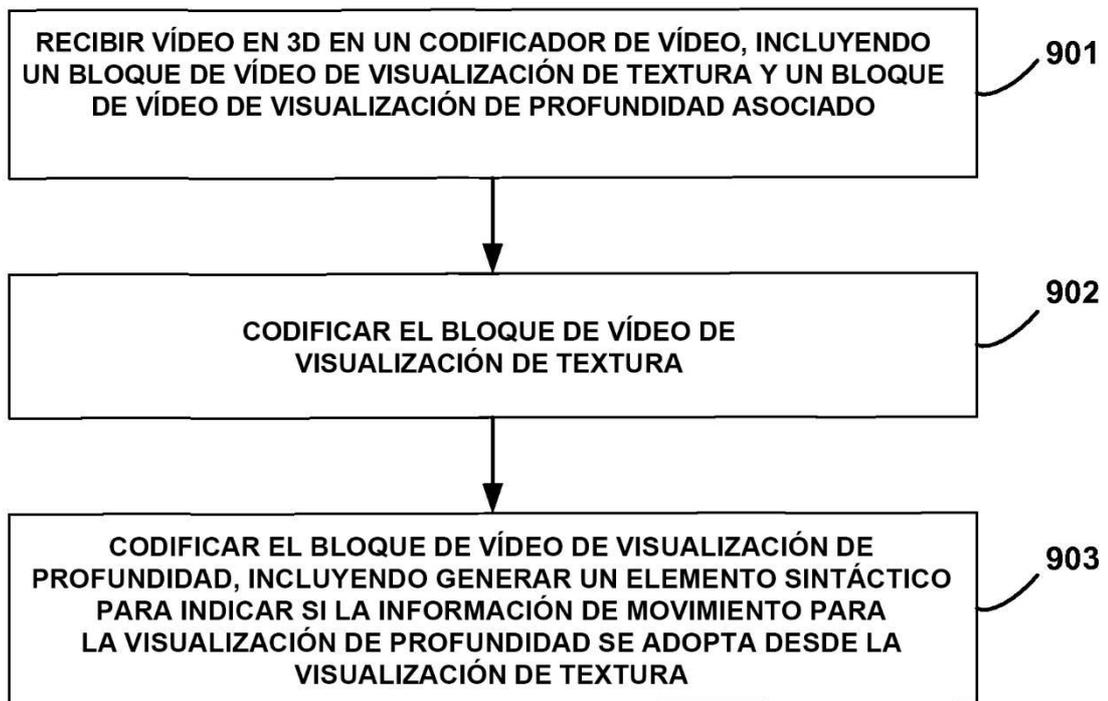


FIG. 9

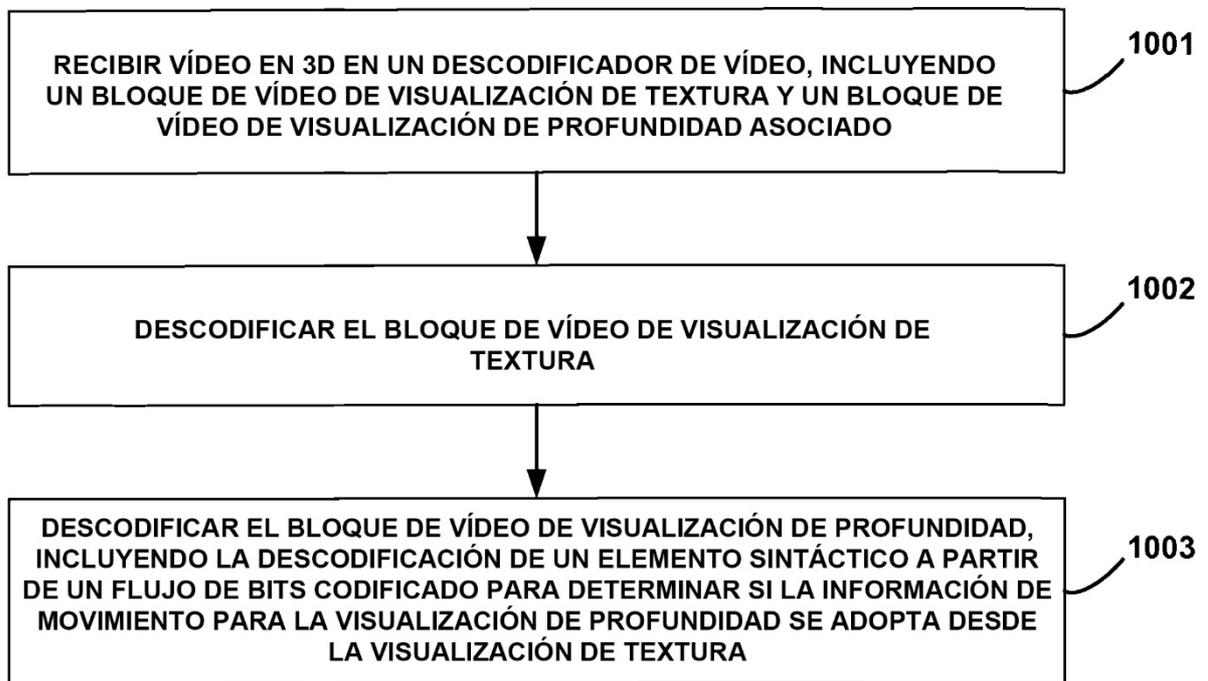


FIG. 10