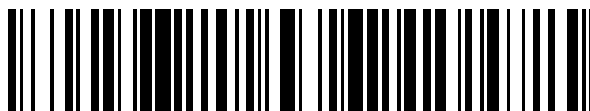


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 693 683**

51 Int. Cl.:

H04N 19/597 (2014.01)

H04N 21/434 (2011.01)

H04N 21/81 (2011.01)

H04N 21/845 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **27.02.2013 PCT/US2013/028050**

87 Fecha y número de publicación internacional: **06.09.2013 WO13130631**

96 Fecha de presentación y número de la solicitud europea: **27.02.2013 E 13709661 (6)**

97 Fecha y número de publicación de la concesión europea: **08.08.2018 EP 2820854**

54 Título: **Extracción de flujo de bits en vídeo tridimensional**

30 Prioridad:

29.02.2012 US 201261605136 P
26.02.2013 US 20131377785

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
13.12.2018

73 Titular/es:

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

72 Inventor/es:

CHEN, YING;
WANG, YE-KUI y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 693 683 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

Extracción de flujo de bits en vídeo tridimensional

5 **CAMPO TÉCNICO**

[0001] Esta divulgación se refiere a la codificación de vídeo (es decir, codificación y/o decodificación de datos de vídeo).

10 **ANTECEDENTES**

[0002] Las capacidades del vídeo digital pueden incorporarse a una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión digital directa, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión de flujo de vídeo y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como las descritas en los estándares definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), el estándar de Codificación de Vídeo de Alta Eficiencia (HEVC) actualmente en desarrollo y las extensiones de dichos estándares. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de compresión de vídeo.

[0003] Las técnicas de compresión de vídeo llevan a cabo la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (es decir, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican mediante predicción espacial con respecto a muestras de referencia en bloques contiguos de la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques contiguos de la misma imagen, o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

[0004] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel a un dominio de transformación, dando como resultado coeficientes residuales, los cuales pueden cuantificarse posteriormente. Los coeficientes cuantificados, inicialmente dispuestos en una matriz bidimensional, pueden escanearse para producir un vector unidimensional de coeficientes, y la codificación por entropía puede aplicarse a los coeficientes cuantificados escaneados para lograr una compresión aún mayor.

[0005] Puede generarse un flujo de bits de múltiples visualizaciones mediante la codificación de visualizaciones por ejemplo, de múltiples cámaras a color. Para ampliar aún más la flexibilidad del vídeo de múltiples visualizaciones, se han desarrollado estándares de vídeo tridimensionales (3D). Un flujo de bits de vídeo 3D puede contener no solo las visualizaciones correspondientes a múltiples cámaras, concretamente visualizaciones de textura, sino también visualizaciones de profundidad asociadas con al menos una o más visualizaciones de textura. Por ejemplo, cada visualización puede consistir en una visualización de textura y una visualización de profundidad.

SUMARIO

[0006] La invención se define mediante las reivindicaciones adjuntas. En general, esta divulgación describe técnicas para extraer un flujo de bits de datos de vídeo a partir de un flujo de bits de vídeo tridimensional (3DV). Más específicamente, un dispositivo determina una lista de visualizaciones de objetivos de textura que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para decodificar imágenes en una pluralidad de visualizaciones de objetivos. Las visualizaciones de objetivos son un subconjunto de las visualizaciones en el flujo de bits que se van a decodificar desde el sub-flujo de bits. Además, el dispositivo determina una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para decodificar imágenes en la pluralidad de visualizaciones de objetivos. El dispositivo determina el sub-flujo de bits basándose, al menos en parte, en la lista de visualizaciones de objetivos de textura y en la lista de visualizaciones de objetivos de profundidad.

[0007] De acuerdo con las técnicas de esta divulgación, una visualización de la profundidad puede excluirse de la lista de visualizaciones de objetivos de profundidad incluso si la visualización de textura asociada a la visualización de la

profundidad se encuentra en la lista de visualizaciones de objetivos de textura. De manera similar, las técnicas de esta divulgación pueden permitir que una visualización de textura, cuando esté disponible, sea excluida de la lista de visualizaciones de objetivos de textura incluso si la visualización de profundidad asociada con la visualización de textura, cuando está disponible, está en la lista de visualizaciones de objetivos de profundidad. Una visualización de textura y una visualización de profundidad se pueden considerar asociadas si corresponden a la misma ubicación de cámara, es decir, en códecs de vídeo 3D como MVC + D o 3D-AVC, que tienen el mismo valor de identificador de visualización (view_id).

[0008] En un ejemplo, esta divulgación describe un procedimiento de extraer un sub-flujo de bits de un flujo de bits 3DV que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados. El procedimiento comprende determinar una lista de visualizaciones de objetivos de textura que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos. Además, el procedimiento comprende determinar una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. Además, el procedimiento comprende determinar el sub-flujo de bits basándose, al menos en parte, en la lista de visualizaciones de objetivos de textura y en la lista de visualizaciones de objetivos de profundidad.

[0009] En otro ejemplo, esta divulgación describe un dispositivo que comprende uno o más procesadores configurados para determinar una lista de visualizaciones de objetivos de textura que indica visualizaciones en un flujo de bits 3DV que tienen componentes de visualización de textura que se requieren para la descodificación de imágenes en una pluralidad de visualizaciones de objetivos, con el flujo de bits 3DV que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados. El uno o más procesadores también están configurados para determinar una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. Además, el uno o más procesadores pueden determinar el sub-flujo de bits basándose al menos en parte en la lista de visualizaciones de objetivos de textura y en la lista de visualizaciones de objetivos de profundidad.

[0010] En otro ejemplo, esta divulgación describe un dispositivo que comprende medios para determinar una lista de visualizaciones de objetivos de textura que indica visualizaciones en un flujo de bits 3DV que tienen componentes de visualización de textura que se requieren para la descodificación de imágenes en una pluralidad de visualizaciones de objetivos, el flujo de bits 3DV incluyendo componentes de visualización de textura codificados y componentes de visualización de profundidad codificados. El dispositivo también comprende medios para determinar una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. Además, el dispositivo comprende medios para determinar el sub-flujo de bits basándose, al menos en parte, en la lista de visualizaciones de objetivos de textura y la lista de visualizaciones de objetivos de profundidad.

[0011] En otro ejemplo, esta divulgación describe un medio de almacenamiento legible por ordenador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores de un dispositivo, configuran el dispositivo para determinar una lista de visualización de objetivos de textura que indica visualizaciones en un flujo de bits 3DV que tiene componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos, con el flujo de bits 3DV que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados. Además, las instrucciones configuran el dispositivo para determinar una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. Además, las instrucciones configuran el dispositivo para determinar el sub-flujo de bits basándose, al menos en parte, en la lista de visualizaciones de objetivos de textura y en la lista de visualizaciones de objetivos de profundidad.

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

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0013]

La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación 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.

5 La FIG. 4 es un diagrama de flujo que ilustra un ejemplo de operación de extracción de sub-flujo de bits, de acuerdo con una o más técnicas de esta divulgación.

10 La FIG. 5 es un diagrama de flujo que ilustra un ejemplo de proceso de extracción de sub-flujo de bits en vídeo tridimensional (3DV) compatible con codificación de múltiples visualizaciones (MVC), de acuerdo con una o más técnicas de esta divulgación.

15 La FIG. 6 es un diagrama de flujo que ilustra una continuación del proceso de extracción de sub-flujo de bits de ejemplo de la FIG. 5.

20 La FIG. 7 es un diagrama de flujo que ilustra una operación de ejemplo para determinar identificadores de visualización de componentes de visualización de textura de anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación.

25 La FIG. 8 es un diagrama de flujo que ilustra una operación de ejemplo para determinar identificadores de visualización de componentes de visualización de profundidad de anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación.

30 La FIG. 9 es un diagrama de flujo que ilustra una operación de ejemplo para determinar identificadores de visualización de componentes de visualización de textura sin anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación.

35 La FIG. 10 es un diagrama de flujo que ilustra una operación de ejemplo para determinar identificadores de visualización de componentes de visualización de profundidad sin anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación.

40 La FIG. 11 es un diagrama de flujo que ilustra una primera operación de ejemplo para marcar unidades de capa de acceso de red (VLC) de capa de codificación de vídeo (NAL) y unidades NAL de datos de relleno para ser eliminadas de un flujo de bits, de acuerdo con una o más técnicas de esta divulgación.

45 La FIG. 12 es un diagrama de flujo que ilustra una segunda operación de ejemplo para marcar unidades VCL NAL y unidades NAL de datos de relleno como para ser eliminadas de un flujo de bits, de acuerdo con una o más técnicas de esta divulgación.

50 La FIG. 13 es un diagrama conceptual que ilustra un ejemplo de orden de descodificación MVC.

55 La FIG. 14 es un diagrama conceptual que ilustra un ejemplo de estructura de predicción entre visualizaciones y temporal de MVC.

DESCRIPCIÓN DETALLADA

60 **[0014]** Un flujo de bits puede incluir datos de vídeo de codificación de vídeo de múltiples visualizaciones (MVC) codificados. Los datos de vídeo MVC pueden incluir datos que definen múltiples visualizaciones de una escena. Para soportar vídeo tridimensional (3DV), usando múltiples visualizaciones más codificación de profundidad de datos de vídeo tridimensionales (3D), al menos algunas de las visualizaciones pueden incluir un componente de textura y un componente de profundidad. Un ejemplo de un proceso de codificación de profundidad más múltiples visualizaciones está en desarrollo como una extensión de MVC bajo el estándar ITU-T H.264. Esta extensión se conoce como la extensión 3DV de H.264/AVC. Un descodificador de vídeo puede descodificar el flujo de bits para dar salida a 3DV. El flujo de bits puede comprender una serie de unidades de capa de abstracción de red (NAL), cada una de las cuales contiene partes de los datos de vídeo MVC codificados. La inclusión de visualizaciones adicionales en los datos de vídeo de múltiples visualizaciones codificados puede aumentar significativamente la velocidad de transmisión de bits del flujo de bits.

65 **[0015]** Algunos dispositivos informáticos que solicitan el flujo de bits no están configurados para manejar todas las visualizaciones incluidas en los datos de vídeo de múltiples visualizaciones. Además, puede haber ancho de banda insuficiente para transmitir el flujo de bits completo a un dispositivo informático que solicita el flujo de bits. En consecuencia, cuando un dispositivo informático solicita el flujo de bits, un dispositivo intermedio, tal como un dispositivo en una red de suministro de contenido (CDN), puede extraer un sub-flujo de bits del flujo de bits. En otras palabras, el dispositivo intermedio puede realizar un proceso de extracción de sub-flujo de bits para extraer el sub-flujo de bits del flujo de bits original. En algunos ejemplos, el dispositivo intermedio extrae el sub-flujo de bits al eliminar selectivamente unidades NAL del flujo de bits original. En otras palabras, la extracción es un proceso en el que las unidades NAL se eliminan y descartan, y las unidades NAL restantes no eliminadas son un sub-flujo de bits. Como resultado, el sub-flujo de bits puede incluir menos visualizaciones que el flujo de bits original. Las visualizaciones que

se incluirán en el sub-flujo de bits se pueden denominar en el presente documento "visualizaciones de objetivos". Para extraer el sub-flujo de bits, el dispositivo intermedio puede eliminar algunas visualizaciones del flujo de bits original, produciendo así el sub-flujo de bits con las visualizaciones restantes. Además, dado que las visualizaciones se eliminan del flujo de bits, la velocidad de transmisión de bits del sub-flujo de bits puede ser menor que la velocidad de transmisión de bits del flujo de bits original, consumiendo menos ancho de banda cuando se envía a través del CDN al dispositivo que solicita el flujo de bits.

[0016] En algunos casos, cada unidad de acceso de los datos de vídeo puede incluir un componente de visualización de textura y un componente de visualización de la profundidad de cada una de las visualizaciones. En esta divulgación, un componente de visualización de textura puede corresponder a un componente de visualización de profundidad, o viceversa, si el componente de visualización de textura y el componente de visualización de profundidad están en la misma visualización y están en la misma unidad de acceso. Si el dispositivo intermedio determina durante el proceso de extracción de sub-flujo de bits que se requiere un componente de visualización de textura para descodificar una imagen en una visualización de objetivos, el dispositivo intermedio no elimina unidades NAL asociadas con el componente de visualización de textura o el componente de visualización de profundidad correspondiente. De forma similar, si el dispositivo intermedio determina durante el proceso de extracción de sub-flujo de bits que se requiere un componente de visualización de profundidad para descodificar una imagen en una visualización de objetivos, el dispositivo intermedio no elimina unidades NAL asociadas con el componente de visualización de profundidad o el componente de visualización de textura correspondiente.

[0017] Sin embargo, hay casos donde se requiere un componente de visualización de textura para descodificar una imagen de una visualización de objetivos, pero no se requiere la correspondiente componente de visualización de profundidad para descodificar cualquier imagen de cualquier visualización de objetivos. De forma similar, hay casos en los que se requiere un componente de visualización de profundidad para descodificar una imagen de una visualización de objetivos, pero no se requiere el componente de visualización de textura correspondiente para descodificar ninguna imagen de ninguna visualización de objetivos. En tales casos, la inclusión tanto del componente de visualización de textura como del componente de visualización de profundidad correspondiente puede ser ineficaz en el ancho de banda. Por consiguiente, puede ser deseable eliminar adicionalmente del flujo de bits original un componente de visualización de profundidad que está asociado con un componente de visualización de textura pero no es necesario descodificar una imagen de una visualización de objetivos o, de forma alternativa, eliminar adicionalmente del flujo de bits original un componente de visualización de textura que está asociado con un componente de visualización de profundidad, pero no es necesario para descodificar una imagen de una visualización de objetivos.

[0018] Las técnicas de esta divulgación están relacionadas con codificación de vídeo MVC y 3D basada en la extensión MVC de H.264/AVC, denominada extensión 3DV de H.264/AVC. Más específicamente, las técnicas de esta divulgación se refieren a la extracción de sub-flujos de un flujo de bits 3DV. El flujo de bits 3DV puede incluir componentes de visualización de textura codificados y componentes de visualización de profundidad codificados. Las técnicas de esta divulgación pueden abordar problemas que ocurren en MVC y 3DV, y ambas pueden requerir que tanto el componente de visualización de textura como el componente de visualización de profundidad correspondiente para una visualización determinada se envíen independientemente de si el componente de visualización de textura y el componente de visualización de profundidad para la visualización dada son realmente necesarios para descodificar una imagen de una visualización de objetivos.

[0019] De acuerdo con las técnicas de esta divulgación, un dispositivo puede mantener, durante un proceso de extracción de sub-flujo de bits, listas de visualizaciones de objetivos independientes para componentes de visualización de textura y componentes de visualización de profundidad. La lista de visualizaciones de objetivos para componentes de visualización de textura puede identificar visualizaciones que tienen componentes de visualización de textura que son necesarios para descodificar una imagen en una visualización de objetivos. La lista de visualizaciones de objetivos para los componentes de visualización de profundidad puede identificar las visualizaciones que tienen componentes de visualización de profundidad que son necesarios para descodificar una imagen en una visualización de objetivos. El dispositivo puede determinar el sub-flujo de bits basándose al menos en parte en las listas de visualizaciones de objetivos para componentes de visualización de textura y profundidad. Por ejemplo, el dispositivo puede eliminar de la capa de codificación de vídeo de flujo de bits (VCL) las unidades NAL que contienen fragmentos codificados de componentes de visualización de textura que pertenecen a visualizaciones no enumeradas en la lista de visualizaciones de objetivos para componentes de visualización de textura. Asimismo, en este ejemplo, el dispositivo puede eliminar del flujo de bits las unidades VCL NAL que contienen fragmentos codificados de componentes de visualización de profundidad que pertenecen a visualizaciones no enumeradas en la lista de visualizaciones de objetivos para componentes de visualización de profundidad.

[0020] Los dibujos adjuntos ilustran ejemplos. Los elementos indicados mediante números de referencia en los dibujos adjuntos corresponden a elementos indicados mediante números de referencia similares en la siguiente descripción. En esta divulgación, los elementos que tienen nombres que comienzan con palabras ordinales (por ejemplo, "primero", "segundo", "tercero", etc.) no necesariamente implican que los elementos tienen un orden particular. Más bien, dichas palabras ordinales se usan simplemente para referirse a diferentes elementos de un mismo tipo o similar.

[0021] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 de ejemplo que puede utilizar las técnicas de esta divulgación. Tal como se utiliza en el presente documento, el término "codificador de vídeo" se refiere genéricamente tanto a codificadores de vídeo como a descodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o la descodificación de vídeo.

[0022] Como se muestra en la FIG. 1, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12, un dispositivo de destino 14 y un dispositivo de red de suministro de contenido (CDN) 16. El dispositivo de origen 12 genera un flujo de bits que incluye datos de vídeo codificados. En consecuencia, el dispositivo de origen 12 puede denominarse dispositivo de codificación de vídeo o aparato de codificación de vídeo. El dispositivo CDN 16 puede extraer un sub-flujo de bits del flujo de bits generado por el dispositivo de origen 12 y puede transmitir el sub-flujo de bits al dispositivo de destino 14. El dispositivo de destino 14 puede descodificar los datos de vídeo codificados en el sub-flujo de bits. En consecuencia, el dispositivo de destino 14 puede denominarse dispositivo de descodificación de vídeo o aparato de descodificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo.

[0023] El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender una amplia variedad de dispositivos, incluyendo ordenadores de sobremesa, dispositivos informáticos móviles, notebooks (es decir, portátiles), ordenadores tipo tablet, descodificadores, equipos telefónicos portátiles tales como los denominados teléfonos "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, ordenadores de coche o similares. El dispositivo CDN 16 puede incluir varios tipos de dispositivos informáticos, tales como dispositivos de servidor, dispositivos informáticos personales, dispositivos de red intermedios (por ejemplo, routers inteligentes, conmutadores, etc.), etc. El dispositivo CDN 16 puede ser parte de una red de suministro de contenido que suministra contenido de vídeo a dispositivos de descodificación de vídeo, tal como el dispositivo de destino 14. Aunque esta divulgación describe un dispositivo CDN como que implementa las técnicas de esta divulgación, las técnicas de esta divulgación pueden implementarse sin el uso de una CDN. Además, las técnicas de esta divulgación pueden implementarse en otros tipos de dispositivos. Por ejemplo, las técnicas de esta divulgación pueden implementarse en el dispositivo de origen 12, el dispositivo destino 14 u otro dispositivo informático.

[0024] El dispositivo CDN 16 puede recibir un flujo de bits desde el dispositivo de origen 12 a través de un canal 17A. El dispositivo de destino 14 puede recibir un sub-flujo de bits desde el dispositivo CDN 16 a través de un canal 17B. Esta divulgación puede referirse colectivamente a los canales 17A y 17B como "canales 17". Cada uno de los canales 17 puede comprender uno o más medios o dispositivos capaces de mover datos de vídeo codificados desde un dispositivo informático a otro dispositivo informático. En un ejemplo, uno o ambos canales 17 pueden comprender uno o más medios de comunicación que permiten a un dispositivo transmitir datos de vídeo codificados directamente a otro dispositivo en tiempo real. En este ejemplo, un dispositivo puede modular los datos de vídeo codificados de acuerdo con un estándar de comunicación, tal como un protocolo de comunicación alámbrico o inalámbrico, y puede transmitir los datos de vídeo modulados a otro dispositivo. Los uno o más medios de comunicación pueden incluir medios de comunicación inalámbricos y/o alámbricos, tales como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. Los uno o más medios de comunicación pueden formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global (por ejemplo, Internet). El uno o más medios de comunicación pueden incluir routers, conmutadores, estaciones base u otros equipos que facilitan la comunicación.

[0025] En otro ejemplo, uno o ambos de los canales 17 puede incluir un medio de almacenamiento que almacene datos de vídeo codificados. En este ejemplo, el dispositivo CDN 16 y/o el dispositivo de destino 14 pueden acceder al medio de almacenamiento a través del acceso al disco o el acceso a la tarjeta. El medio de almacenamiento puede incluir varios medios de almacenamiento de datos de acceso local, tales como discos Blu-ray, DVD, CD-ROM, memoria flash u otros medios adecuados de almacenamiento digital para almacenar datos de vídeo codificados.

[0026] En un ejemplo adicional, cualquiera o ambos de los canales 17 puede incluir un servidor de archivos u otro dispositivo de almacenamiento intermedio que almacene datos de vídeo codificados. En este ejemplo, el dispositivo CDN 16 y/o el dispositivo de destino 14 pueden acceder a los datos de vídeo codificados almacenados en el servidor de archivos u otro dispositivo de almacenamiento intermedio mediante transmisión o descarga. El servidor de ficheros puede ser un tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir los datos de vídeo codificados al dispositivo de destino 14. Entre los ejemplos de servidores de ficheros se incluyen servidores de la Red (por ejemplo, para una sede de la Red), servidores del protocolo de transferencia de ficheros (FTP), dispositivos de almacenamiento conectados a red (NAS) y unidades de disco local.

[0027] El dispositivo CDN 16 y/o el dispositivo de destino 14 pueden acceder a los datos de vídeo codificados a través de una conexión de datos estándar, tal como una conexión a Internet. Entre los ejemplos de tipos de conexiones de datos pueden incluirse canales inalámbricos (por ejemplo, conexiones de Wi-Fi), conexiones cableadas (por ejemplo, DSL, módem de cable, etc.), o combinaciones de ambos que sean adecuadas para acceder a datos de vídeo codificados almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificados desde el servidor de ficheros puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0028] Las técnicas de esta divulgación no están limitadas a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, en soporte de una diversidad de aplicaciones multimedia, tales como radiodifusiones de televisión por el aire, radiodifusiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, mediante Internet, codificación de datos de vídeo para su almacenamiento en un medio de almacenamiento de datos, descodificación de datos de vídeo almacenados en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema de codificación de vídeo puede configurarse para soportar la transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como la transmisión de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0029] 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 ejemplos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. La fuente de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una cámara de vídeo, un archivo de vídeo que contiene datos de vídeo previamente capturados, una interfaz de alimentación de vídeo para recibir datos de vídeo de un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de vídeo, una combinación de tales fuentes de datos de vídeo u otra fuente o fuentes de datos de vídeo.

[0030] El codificador de vídeo 20 puede codificar datos de vídeo de la fuente de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 transmite directamente los datos de vídeo codificados a través de la interfaz de salida 22. En otros ejemplos, los datos de vídeo codificados también pueden almacenarse en un medio de almacenamiento o un servidor de ficheros para un acceso posterior mediante el dispositivo de destino 14 para su descodificación y/o reproducción.

[0031] En el ejemplo de la FIG. 1, el dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 32. En algunos ejemplos, la interfaz de entrada 28 incluye un receptor y/o un módem. La interfaz de entrada 28 puede recibir datos de vídeo codificados a través del canal 17B. El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En general, el dispositivo de visualización 32 muestra datos de vídeo descodificados. El dispositivo de visualización 32 puede comprender varios 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.

[0032] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 funcionan de acuerdo con un estándar de compresión de vídeo, tal como ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluidas sus extensiones de codificación de vídeo escalable (SVC) y codificación de vídeo de múltiples visualizaciones (MVC). A partir del 26 de febrero de 2013, se puede descargar un borrador reciente de la extensión MVC de H.264/AVC en http://wftp3.itu.int/av-arch/jvt-site/2009_01_Geneva/JVT-AD007.zip. Otro borrador conjunto de MVC se describe en "Advanced Video Coding for Generic Audiovisual Services [Codificación de vídeo avanzada para servicios audiovisuales genéricos]", Recomendación ITU-T H.264, marzo de 2010. Además, hay un estándar de codificación de vídeo, como una extensión del estándar MVC, concretamente "MVV basado en 3DV", descrito en Suzuki et al., "WD of MVC extension for inclusion of depth maps [Extensión WD de MVC para la inclusión de mapas de profundidad]", ISO/IEC/JTC1/SC29/WG11/N12351, diciembre de 2011 (en adelante, "documento MPEG w12351").

[0033] En otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 puede funcionar de acuerdo con otros estándares de compresión de vídeo, incluyendo el estándar H.265/de codificación de vídeo de alta eficiencia (HEVC). Un borrador del siguiente estándar H.265/HEVC, denominado "Borrador de Trabajo 9 de la HEVC" se describe en el documento de Bross et al. "High Efficiency Video Coding (HEVC) text specification draft 9 [Borrador 9 de Memoria Descriptiva Textual de Codificación de Vídeo de Alta Eficacia (HEVC)]", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, 11.^a reunión: Shanghái, China, octubre de 2012, que, desde el 26 de febrero de 2013, se puede descargar en: http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v8.zip. Entre otros estándares de codificación de vídeo de ejemplo se incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263 e ISO/IEC MPEG-4 Visual. Aunque las técnicas de esta divulgación se describen con respecto a H.264/AVC, y como ejemplo, con la extensión 3DV de H.264/AVC usando MVC más técnicas de profundidad, tales técnicas también se pueden aplicar a otros estándares de codificación que hacen uso de técnicas de MVC más profundidad, y no están limitadas a ningún estándar o técnica de codificación en particular.

[0034] La FIG. 1 es meramente un ejemplo y las técnicas de esta divulgación pueden aplicarse a ajustes de codificación de vídeo (por ejemplo, codificación de vídeo o descodificación de vídeo) que no incluyen necesariamente ninguna comunicación de datos entre los dispositivos de codificación y descodificación. En otros ejemplos, los datos se recuperan de una memoria local, se transmiten por flujo a través de una red o similar. Un dispositivo de codificación de vídeo puede codificar y almacenar datos en la memoria, y/o un dispositivo de descodificación de vídeo puede recuperar y descodificar datos desde la memoria. En muchos ejemplos, la codificación y descodificación se realiza mediante dispositivos que no se comunican entre sí, sino que simplemente codifican datos en la memoria y/o recuperan y descodifican datos desde la memoria.

[0035] El codificador de vídeo 20, el descodificador de vídeo 30 y el dispositivo CDN 16 pueden implementarse como cualquiera de una variedad de circuitos adecuados, tales como uno o más microprocesadores, procesadores de señal digital (DSP), circuitos integrados específicos de la aplicación (ASIC) , matrices de puertas programables sobre el terreno (FPGA), lógica discreta, hardware o cualquier combinación de los mismos. Si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado de almacenamiento no transitorio legible por ordenador, y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cualquiera de los anteriores (incluyendo hardware, software, una combinación de hardware y software, etc.) puede considerarse como uno o más procesadores. 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.

[0036] Esta divulgación puede referirse en general al codificador de vídeo 20 que "señala" cierta información a otro dispositivo, tal como el descodificador de vídeo 30. El término "indicar" puede referirse en general a la comunicación de elementos sintácticos y/u otros datos usados para descodificar los datos de vídeo comprimidos. Dicha comunicación puede producirse en tiempo real o casi real. De forma alternativa, dicha comunicación puede producirse durante un tramo de tiempo, tal como podría ocurrir cuando se almacenan elementos sintácticos en un medio de almacenamiento legible por ordenador en un flujo de bits codificado en el momento de la codificación, que a continuación un dispositivo de descodificación puede recuperar en cualquier momento tras haber sido almacenado en este medio. Por consiguiente, la señalización puede referirse en general a la provisión de información en un flujo de bits codificado para uso en el procesamiento y/o la descodificación del flujo de bits codificado.

[0037] Una secuencia de vídeo incluye típicamente una serie de tramas de vídeo. Un grupo de imágenes (GOP) comprende en general una serie de una o más tramas de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, una cabecera de una o más tramas del GOP o en otras ubicaciones, que indican el número de tramas incluidas en el GOP. Cada trama puede incluir datos sintácticos de trama que describen un modo de codificación para la respectiva trama. Un codificador de vídeo 20 actúa típicamente sobre unidades de vídeo de tramas de vídeo individuales con el fin de codificar los datos de vídeo. En H.264/AVC, una unidad de vídeo puede corresponder a un macrobloque o una partición de un macrobloque. Un MB es un bloque 16x16 de muestras de luma y dos bloques correspondientes de muestras de croma de una imagen que tiene tres matrices de muestras, o un bloque 16x16 de muestras de una imagen monocromática o una imagen que se codifica mediante tres planos de color independientes. Una partición de MB es un bloque de muestras de luma y dos bloques correspondientes de muestras de croma resultantes de una partición de un macrobloque para la predicción inter para una imagen que tiene tres matrices de muestras o un bloque de muestras de luma resultante de una partición de un macrobloque para predicción inter de una imagen monocromática o una imagen que se codifica mediante tres planos de color independientes. En H.265/HEVC, una unidad de vídeo puede corresponder a una unidad de predicción (PU). Una PU puede ser un bloque de predicción de muestras de luma, dos bloques de predicción correspondientes de muestras de croma de una imagen y estructuras sintácticas usadas para predecir las muestras de bloques de predicción. Un bloque de predicción puede ser un bloque rectangular de muestras (por ejemplo, MxN) en el que se aplica la misma predicción. Los bloques de unidades de vídeo pueden tener tamaños fijos o variables, y pueden diferir en tamaño de acuerdo con un estándar de codificación específico.

[0038] Cuando el codificador de vídeo 20 codifica una unidad de vídeo actual, el codificador de vídeo 20 puede generar bloques predictivos que se corresponden con la unidad de vídeo actual. El codificador de vídeo 20 puede realizar predicción intra o predicción inter para generar los bloques predictivos. Cuando el codificador de vídeo 20 realiza la predicción intra para una unidad de vídeo actual, el codificador de vídeo 20 puede generar, basándose en muestras de la misma imagen que la unidad de vídeo actual, bloques predictivos para la unidad de vídeo actual. Cuando el codificador de vídeo 20 realiza predicción inter para generar bloques predictivos para una unidad de vídeo actual, el codificador de vídeo 20 puede generar los bloques predictivos basándose en muestras de una o más imágenes de referencia. Las imágenes de referencia pueden ser imágenes distintas de la imagen que contiene la unidad de vídeo actual.

[0039] Una vez que el codificador de vídeo 20 ha generado bloques predictivos para una unidad de vídeo actual, el codificador de vídeo 20 puede generar bloques residuales para la unidad de vídeo actual. Cada muestra de un bloque residual puede basarse en una diferencia entre unas muestras correspondientes de un bloque de luma o croma de la unidad de vídeo actual y un bloque predictivo para la unidad de vídeo actual. El codificador de vídeo 20 puede aplicar una transformación a unas muestras de un bloque residual para generar un bloque de coeficientes de transformación. El codificador de vídeo 20 puede aplicar varias transformaciones al bloque residual. Por ejemplo, el codificador de vídeo 20 puede aplicar una transformación tal como una transformación de coseno discreta (DCT), una transformación de número entero, una transformación de tren de ondas o una transformación conceptualmente similar al bloque residual. El codificador de vídeo 20 puede cuantificar los bloques de coeficientes de transformación para reducir aún más el número de bits usados para representar la unidad de vídeo actual. Después de cuantificar un bloque de coeficientes de transformación, el codificador de vídeo 20 puede codificar por entropía los elementos sintácticos que representan los coeficientes de transformación en el bloque de coeficientes de transformación y otros elementos sintácticos asociados con la unidad de vídeo actual. Por ejemplo, el codificador de vídeo 20 puede realizar codificación aritmética binaria adaptable al contexto (CABAC), codificación de longitud variable adaptable al contexto (CAVLC),

codificación de Golomb exponencial u otro tipo de codificación por entropía en los elementos sintácticos. El codificador de vídeo 20 puede emitir un flujo de bits que incluye los elementos sintácticos codificados por entropía asociados con la unidad de vídeo actual.

5 **[0040]** El flujo de bits puede incluir una secuencia de bits que forma una representación de imágenes codificadas y datos asociados. El flujo de bits puede comprender una secuencia de unidades de capa de abstracción de red (NAL). Cada una de las unidades de NAL incluye una cabecera de la unidad NAL y encapsula una carga útil de secuencia de octetos sin procesar (RBSP). La cabecera de la unidad NAL puede incluir un elemento sintáctico que indica un código de tipo de unidad NAL. El código de tipo de unidad NAL especificado por la cabecera de unidad NAL de una unidad
10 NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contiene un número entero de bytes que se encapsula dentro de una unidad NAL. En algunos casos, una RBSP incluye cero bits.

[0041] Diferentes tipos de unidades NAL pueden encapsular diferentes tipos de RBSP. Por ejemplo, un primer tipo de unidad NAL puede encapsular una RBSP para un conjunto de parámetros de imagen (PPS), un segundo tipo de unidad
15 NAL puede encapsular una RBSP para un fragmento codificado, un tercer tipo de unidad NAL puede encapsular una RBSP para información de realce complementaria (SEI), etc. Las unidades NAL que encapsulan las RBSP para datos de codificación de vídeo (a diferencia de las RBSP para conjuntos de parámetros y mensajes SEI) pueden denominarse unidades NAL de la capa de codificación de vídeo (VCL). Una unidad NAL que encapsula un fragmento codificado puede denominarse unidad NAL de fragmento codificado.

[0042] El descodificador de vídeo 30 puede recibir un flujo de bits que incluye una representación codificada de datos de vídeo. Por ejemplo, el descodificador de vídeo 30 puede analizar el flujo de bits para extraer elementos sintácticos del flujo de bits. Como parte de la extracción de los elementos sintácticos del flujo de bits, el descodificador de vídeo
20 30 puede realizar la descodificación por entropía de partes del flujo de bits. El descodificador de vídeo 30 puede realizar, basándose al menos en parte en los elementos sintácticos asociados con una unidad de vídeo actual (por ejemplo, una partición MB o MB), predicción inter o intra para generar bloques predictivos para la unidad de vídeo actual. Además, el descodificador de vídeo 30 puede cuantificar inversamente los coeficientes de transformación de los bloques de coeficientes de transformación asociados con la unidad de vídeo actual y puede aplicar una o más transformaciones inversas a los bloques de coeficientes de transformación para generar bloques residuales para la
25 unidad de vídeo actual. El descodificador de vídeo 30 puede entonces reconstruir los bloques de luma y croma de la unidad de vídeo actual basándose, al menos en parte, en los bloques residuales y los bloques predictivos. De este modo, mediante la reconstrucción de los bloques de luma y croma de cada unidad de vídeo de una imagen, el descodificador de vídeo 30 puede reconstruir la imagen.

[0043] Como se ha mencionado anteriormente, el codificador de vídeo 20 puede llevar a cabo la predicción inter para generar bloques predictivos para una unidad de vídeo particular. Más específicamente, el codificador de vídeo 20
30 puede realizar una predicción inter unidireccional o una predicción inter bidireccional para generar los bloques predictivos.

[0044] Cuando el codificador de vídeo 20 realiza predicción inter unidireccional para una unidad de vídeo actual (por ejemplo, un MB o partición MB), el codificador de vídeo 20 pueden buscar un bloque de referencia dentro de imágenes de referencia en una única lista de imágenes de referencia. El bloque de referencia puede ser un bloque de muestras de luma y unos bloques correspondientes de muestras de croma que son similares a los bloques de luma y croma de la unidad de vídeo actual. Además, cuando el codificador de vídeo 20 realiza una predicción inter unidireccional, el
35 codificador de vídeo 20 puede generar información de movimiento para el bloque de vídeo particular. La información de movimiento para la unidad de vídeo particular puede incluir un vector de movimiento y un índice de referencia. El vector de movimiento puede indicar un desplazamiento espacial entre una posición de la imagen actual de los bloques de la unidad de vídeo actual y una posición de la imagen de referencia del bloque de referencia. El índice de referencia indica una posición de la lista de imágenes de referencia de la imagen de referencia que contiene el bloque de referencia. Las muestras de los bloques predictivos para la unidad de vídeo actual pueden ser iguales a unas muestras correspondientes del bloque de referencia.

[0045] Cuando el codificador de vídeo 20 realiza predicción inter bidireccional para una unidad de vídeo actual (por ejemplo, un MB o partición MB), el codificador de vídeo 20 puede buscar un primer bloque de referencia dentro de
40 imágenes de referencia en una primera lista de imágenes de referencia ("lista 0") y puede buscar un segundo bloque de referencia dentro de imágenes de referencia en una segunda lista de imágenes de referencia ("lista 1"). El codificador de vídeo 20 puede generar, basándose al menos en parte en el primer y el segundo bloques de referencia, los bloques predictivos para la unidad de vídeo actual. Además, el codificador de vídeo 20 puede generar un primer vector de movimiento que indica un desplazamiento espacial entre los bloques de la unidad de vídeo actual y el primer
45 bloque de referencia. El codificador de vídeo 20 también puede generar un primer índice de referencia que identifica una ubicación de la primera lista de imágenes de referencia de la imagen de referencia que contiene el primer bloque de referencia. Además, el codificador de vídeo 20 puede generar un segundo vector de movimiento que indica un desplazamiento espacial entre los bloques de la unidad de vídeo actual y el segundo bloque de referencia. El codificador de vídeo 20 también puede generar un segundo índice de referencia que identifica una ubicación de la
50 segunda lista de imágenes de referencia de la imagen de referencia que incluye el segundo bloque de referencia.

[0046] Cuando el codificador de vídeo 20 realiza predicción inter unidireccional para una unidad de vídeo actual (por ejemplo, un MB o partición MB), el descodificador de vídeo 30 puede usar la información de movimiento de la unidad de vídeo actual para identificar el bloque de referencia de la unidad de vídeo actual. El descodificador de vídeo 30 puede generar a continuación los bloques predictivos para la unidad de vídeo actual basándose en el bloque de referencia de la unidad de vídeo actual. Cuando el codificador de vídeo 20 realiza la predicción inter bidireccional para la unidad de vídeo actual, el descodificador de vídeo 30 puede utilizar la información de movimiento para la unidad de vídeo actual para identificar los dos bloques de referencia de la unidad de vídeo actual. El descodificador de vídeo 30 puede generar los bloques predictivos de la unidad de vídeo actual basándose en los dos bloques de referencia de la unidad de vídeo actual.

[0047] La codificación de vídeo de múltiples visualizaciones (MVC) es una extensión del estándar H.264/AVC. La divulgación describe técnicas para vídeo tridimensional (3DV), usando codificación MVC más profundidad de datos de vídeo tridimensionales (3D), como en la extensión 3DV de H.264/AVC. Ahora se presentará un breve análisis de MVC. En la extensión de MVC para H.264, puede haber múltiples visualizaciones de la misma escena desde diferentes visualizaciones. El término "unidad de acceso" se usa para referirse al conjunto de componentes de visualización que corresponden a la misma instancia de tiempo. Un "componente de visualización" puede ser una representación codificada de una visualización en una única unidad de acceso. Así, los datos de vídeo pueden conceptualizarse como una serie de unidades de acceso que se producen a lo largo del tiempo. En esta divulgación, una "visualización" puede referirse a una secuencia de componentes de visualización asociadas con el mismo identificador de visualización. Las unidades VCL NAL que contienen fragmentos codificados de componentes de visualización que pertenecen a la misma visualización especifican el mismo identificador de visualización. Un "índice de orden de visualización" es un índice que indica el orden de descodificación de componentes de visualización en una unidad de acceso.

[0048] MVC soporta predicción entre visualizaciones. La predicción entre visualizaciones es similar a la predicción inter utilizada en H.264/AVC y puede usar los mismos elementos sintácticos. Sin embargo, cuando un codificador de vídeo realiza predicción entre visualizaciones en una unidad de vídeo actual, el codificador de vídeo 20 puede usar, como una imagen de referencia, una imagen que está en la misma unidad de acceso que la unidad de vídeo actual, pero en una visualización diferente. En contraste, la predicción inter convencional solo usa imágenes en diferentes unidades de acceso como imágenes de referencia. En MVC, una visualización se denomina "visualización básica" si un descodificador de vídeo (por ejemplo, descodificador de vídeo 30) puede descodificar cada imagen de la visualización sin referencia a imágenes de ninguna otra visualización. Al codificar una imagen en una de las visualizaciones no básicas, un codificador de vídeo puede agregar una imagen en una lista de imágenes de referencia si la imagen está en una visualización diferente pero dentro de una misma instancia de tiempo (es decir, unidad de acceso) que la imagen que el codificador de vídeo actualmente está codificando. Al igual que otras imágenes de referencia de predicción inter, el codificador de vídeo puede insertar una imagen de referencia de predicción entre visualizaciones en cualquier posición de una lista de imágenes de referencia. En MVC, la predicción entre visualizaciones puede ser soportada por la compensación de movimiento de disparidad. La compensación de movimiento de disparidad usa la sintaxis de la compensación de movimiento H.264/AVC, pero puede permitir que una imagen en una visualización diferente se use como una imagen de referencia. La codificación de dos o más visualizaciones puede ser soportada por MVC. Una de las ventajas de MVC es que un codificador MVC puede usar más de dos visualizaciones como una entrada de vídeo 3D y un descodificador MVC puede descodificar dicha representación de múltiples visualizaciones. Como resultado, los descodificadores de vídeo que soportan MVC pueden procesar contenido de vídeo 3D con más de dos visualizaciones.

[0049] Además, como se mencionó anteriormente, hay una extensión 3DV compatible con MVC emergente de H.264/AVC. 3DV compatible con MVC está diseñada para permitir mejoras 3D manteniendo la compatibilidad MVC. 3DV compatible con MVC proporciona mapas de profundidad. En consecuencia, se puede hacer referencia a 3DV compatible con MVC como "MVC más profundidad", "MVC + D" o como "extensión compatible con MVC que incluye profundidad". Un borrador reciente de 3DV compatible con MVC se proporciona en Suzuki et al., "WD on MVC extensions for inclusion of depth maps [WD en extensiones MVC para la inclusión de mapas de profundidad]", ISO/IEC/JTC1/SC29/WG11/N12351, diciembre de 2011, es un borrador de 3DV compatible con MVC. Suzuki et al., "WD en extensiones MVC para la inclusión de mapas de profundidad", ISO/IEC/JTC1/SC29/WG11/N12544, febrero de 2012(en lo sucesivo denominado "documento N12544").

[0050] Un componente de visualización de profundidad incluye un mapa de profundidad. Los mapas de profundidad son imágenes cuyos valores de píxel representan las profundidades tridimensionales de los objetos que se muestran en las correspondientes imágenes de "textura". En algunos ejemplos, los valores de píxeles más brillantes en un mapa de profundidad pueden corresponder a objetos que están más cerca de una cámara y los valores de píxeles más oscuros en un mapa de profundidad pueden corresponder a objetos que están más alejados de la cámara. Las imágenes del componente de "textura" pueden ser imágenes H.264/AVC normales.

[0051] En esta divulgación, se puede aplicar la terminología de 3DV compatible con MVC. Por ejemplo, la parte de textura de una visualización se puede denominar una "visualización de textura" y la parte de profundidad de una visualización se puede denominar una "visualización de profundidad". La parte de textura de una visualización en una unidad de acceso, es decir, una visualización de textura en una unidad de acceso, se puede denominar "componente de visualización de textura". La parte de profundidad de una visualización en una unidad de acceso, es decir, una

visualización de profundidad en una unidad de acceso, se puede denominar un "componente de visualización de profundidad". El término "componente de visualización" puede usarse para referirse a una visualización en una unidad de acceso y colectivamente tanto al componente de visualización de textura como al componente de visualización de profundidad de la misma unidad de acceso.

5 **[0052]** El codificador de vídeo 20 puede usar la renderización basada en la imagen de profundidad (DIBR) para generar, basándose en los componentes de visualización de textura y profundidad disponibles, un componente de visualización de textura sintética. Un componente de visualización de textura sintética puede ser un componente de visualización de textura que se sintetiza basándose en un mapa de profundidad y uno o más componentes de visualización de textura. Por ejemplo, un componente de visualización de textura particular puede ser un componente de visualización de textura de ojo izquierdo y el codificador de vídeo 20 puede usar DIBR para generar un componente de visualización de textura de ojo derecho para la reproducción de vídeo tridimensional. En algunos casos, se puede usar un componente de visualización de textura sintética como una imagen de referencia para la predicción de la unidad entre accesos o la predicción entre visualizaciones. Los componentes de visualización de textura sintética que se usan como imágenes de referencia se pueden denominar imágenes de referencia de síntesis de visualización (VSRP). Los codificadores de vídeo pueden incluir VSRP en listas de imágenes de referencia.

20 **[0053]** Como se ha analizado anteriormente, un componente de visualización puede utilizar los componentes de visualización en otras visualizaciones para referencia; esto se llama predicción entre visualizaciones. En MVC (y 3DV compatible con MVC), la predicción entre visualizaciones puede implementarse como si el componente de visualización en otra visualización fuera una imagen de referencia entre predicciones. Las posibles imágenes de referencia entre visualizaciones pueden señalarse en una extensión SPS para MVC (es decir, una extensión SPS MVC). La Tabla 1, a continuación, muestra una sintaxis de ejemplo para la extensión SPS MVC para H.264.

25 **TABLA 1**

		Descriptor
1	conjunto de parámetros seq extensión mvc() {	
2	num_views_minus1	ue(v)
3	for(i = 0; i <= num_views_minus1; i++)	
4	view_id[i]	ue(v)
5	for(i = 1; i <= num_views_minus1; i++) {	
6	num_anchor_refs_l0[i]	ue(v)
7	for(j = 0; j < num_anchor_refs_l0[i]; j++)	
8	anchor_ref_l0[i][j]	ue(v)
9	num_anchor_refs_l1[i]	ue(v)
10	for(j = 0; j < num_anchor_refs_l1[i]; j++)	
11	anchor_ref_l1[i][j]	ue(v)
12	}	
13	for(i = 1; i <= num_views_minus_1; i++) {	
14	num_non_anchor_refs_l0[i]	ue(v)
15	for(j = 0; j < num_non_anchor_refs_l0[i]; j++)	
16	non_anchor_ref_l0[i][j]	ue(v)
17	num_non_anchor_refs_l1[i]	ue(v)
18	for(j = 0; j < num_non_anchor_refs_l1[i]; j++)	
19	non_anchor_ref_l1[i][j]	ue(v)
20	}	
21	num_level_values_signalled_minus1	ue(v)
22	for(i = 0; i <= num_level_values_signalled_minus1; i++) {	
23	level_idc[i]	u(8)
24	num_applicable_ops_minus1[i]	ue(v)
25	for(j = 0; j <= num_applicable_ops_minus1[i]; j++) {	

1	conjunto de parámetros seq extensión mvc() {	Descriptor
26	applicable_op_temporal_id [i][j]	u(3)
27	applicable_op_num_target_views_minus1 [i][j]	ue(v)
28	for(k = 0; k <= applicable_op_num_target_views_minus1[i][j]; k++)	
29	applicable_op_target_view_id [i][j][k]	ue(v)
30	applicable_op_num_views_minus1 [i][j]	ue(v)
31	}	
32	}	
33	}	

[0054] En la Tabla 1 anterior y en otras tablas sintácticas de esta divulgación, los elementos sintácticos con descriptor de tipo ue(v) pueden ser números enteros de longitud variable sin signo codificados utilizando codificación Golomb exponencial (Exp-golomb) de orden 0 con el bit izquierdo primero. En el ejemplo de la Tabla 1 y las siguientes tablas, un elemento sintáctico que tiene un descriptor de la forma u(n), donde n es un entero no negativo, son valores sin signo de longitud n. Los elementos sintácticos con los descriptores de tipo u(3) y u(8) pueden ser enteros sin signo con 3 y 8 bits, respectivamente.

[0055] Como se muestra en las líneas 3 y 4 de la Tabla 1, la extensión SPS MVC puede especificar identificadores de visualización (por ejemplo, view_id[i]) de visualizaciones aplicables. Además, el codificador de vídeo 20 puede, en la extensión SPS MVC de la Tabla 1, señalar, para cada visualización, el número de visualizaciones que pueden usarse para formar la lista de imágenes de referencia 0 y la lista de imágenes de referencia 1. Por ejemplo, los elementos sintácticos num_anchor_refs_0[i] y num_non_anchor_refs_0[i] en las líneas 6 y 14 de la Tabla 1 pueden especificar el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial 0 en la descodificación de componentes de visualización de anclaje y sin anclaje, respectivamente, con índice de orden de visualización igual a i. Los elementos sintácticos num_anchor_refs_1[i] y num_non_anchor_refs_1[i] en las líneas 9 y 17 de la Tabla 1 pueden especificar el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial 1 en la descodificación de componentes de visualización de anclaje y sin anclaje, respectivamente, con índice de orden de visualización igual a i.

[0056] La extensión SPS MVC puede especificar posibles dependencias entre visualizaciones y visualizar componentes aplicables al SPS. Por ejemplo, en el ejemplo de la Tabla 1, la extensión SPS MVC puede incluir elementos sintácticos denotados anchor_ref_0[i][j], anchor_ref_1[i][j], non_anchor_ref_0[i][j] y non_anchor_ref_1[i][j]. Anchor_ref_0[i][j] especifica el identificador de visualización (identificación de visualización) del j-ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 en la descodificación de componentes de visualización de anclaje con índice de orden de visualización (VOIdx) igual a i. Anchor_ref_1[i][j] especifica el identificador de visualización (view_id) del j-ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList1 en la descodificación de componentes de visualización de anclaje con índice de orden de visualización (VOIdx) igual a i. Non_anchor_ref_0[i][j] especifica el identificador de visualización (view_id) del j-ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 en la descodificación de componentes de visualización sin anclaje con índice de orden de visualización (VOIdx) igual a i. Anchor_ref_1[i][j] especifica el identificador de visualización (view_id) del j-ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList1 en la descodificación de componentes de visualización sin anclaje con el índice de orden de visualización (VOIdx) igual a i. De esta forma, la extensión SPS MVC puede especificar las posibles relaciones de predicción (es decir, dependencias) entre los componentes de visualización de anclaje y las posibles imágenes de referencia entre visualizaciones. Además, de esta forma, la extensión SPS MVC puede especificar las posibles relaciones de predicción (es decir, dependencias) entre componentes de visualización sin anclaje y posibles imágenes de referencia entre visualizaciones.

[0057] Una relación de predicción para una imagen de anclaje, tal como se señala en la extensión de la MVC del SPS, puede ser diferente a la relación de predicción para una imagen sin anclaje (señalada en la extensión de la MVC del SPS) de la misma visualización. Las listas de imágenes de referencia pueden modificarse durante el proceso de construcción de la lista de imágenes de referencia para permitir el ordenamiento flexible de las imágenes de referencia de predicción inter o predicción entre visualizaciones en las listas de imágenes de referencia.

[0058] En MVC (y 3DV compatible con MVC), una unidad NAL puede incluir una cabecera de unidad NAL de un byte y una extensión de cabecera de unidad MVC NAL de tres bytes si la unidad NAL es una unidad NAL prefijo o una unidad MVC VCL NAL. La cabecera de la unidad NAL de un byte puede incluir un tipo de unidad NAL y un elemento sintáctico nal_ref_idc. El elemento sintáctico nal_ref_idc especifica si la unidad NAL contiene un SPS, una extensión

SPS, un subconjunto SPS, un PPS, un fragmento de una imagen de referencia, una partición de datos de fragmento de una imagen de referencia o una unidad NAL prefijo que precede a un fragmento de una imagen de referencia. Una unidad NAL de prefijo en MVC (y 3DV compatible con MVC) puede contener solo una cabecera de unidad NAL y la extensión de cabecera de unidad MVC NAL. La Tabla 2 siguiente indica una estructura sintáctica de ejemplo para la extensión de cabecera de unidad MVC NAL (nal_unit_header_mvc_extension).

TABLA 2

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

10 **[0059]** En la estructura sintáctica de ejemplo de la Tabla 2, la non_idr_flag indica si la unidad NAL pertenece a una
 unidad NAL de actualización de descodificación instantánea (IDR) que puede ser utilizada como un punto de acceso
 aleatorio GOP cerrado. Un punto de acceso aleatorio es una imagen que incluye solo 1 fragmentos. El elemento
 sintáctico priority_id se puede usar para adaptación de una vía, en el que la adaptación se puede hacer simplemente
 comprobando priority_id. El elemento sintáctico view_id puede indicar un identificador de visualización de una
 15 visualización actual. La unidad NAL puede encapsular una representación codificada de un fragmento de un
 componente de visualización de la visualización actual. El elemento sintáctico temporal_id puede indicar un nivel
 temporal de la unidad NAL. El nivel temporal puede indicar una velocidad de trama asociada con la unidad NAL. El
 elemento sintáctico anchor_pic_flag puede indicar si la unidad NAL pertenece a una imagen de anclaje que se puede
 usar como punto de acceso aleatorio GOP abierto. Una imagen de anclaje es una imagen codificada en la que todos
 20 los fragmentos solo pueden hacer referencia a los fragmentos en la misma unidad de acceso. Es decir, la predicción
 entre visualizaciones se puede usar para codificar una imagen de anclaje, pero la predicción inter no se puede usar
 para codificar la imagen de anclaje. El elemento sintáctico inter_view_flag indica si un componente de visualización
 actual se utiliza para predicción entre visualizaciones para unidades NAL en otras visualizaciones. La unidad NAL
 puede encapsular una representación codificada de un fragmento del componente de visualización actual.

25 **[0060]** La solicitud de patente de Estados Unidos 13/414 515, presentada el 7 de marzo 2012 introduce un elemento
 sintáctico depth_to_view_flag en la cabecera de unidad NAL para indicar si el componente de visualización actual, si
 se trata de un componente de visualización de textura, no se utiliza para predecir cualquier componente de
 visualización profundidad. En otras palabras, si la unidad NAL encapsula un fragmento codificado de un componente
 30 de visualización de textura, un elemento sintáctico depth_to_view_flag en la cabecera de unidad NAL de la unidad NAL
 indica si el componente de visualización de textura se usa para predecir un componente de visualización de
 profundidad.

35 **[0061]** Con la aparición de las redes de distribución de contenidos (CDN) para el suministro de contenido de 3D de
 múltiples visualizaciones a través de Internet u otras redes de datos, 3DV compatible con MVC se ha adaptado para
 facilitar el suministro de los flujos de bits de datos de vídeo que incluyen todas las diversas visualizaciones a los CDN.
 Por ejemplo, un servidor de biblioteca de vídeo central (u otros dispositivos) puede codificar datos de vídeo de múltiples
 visualizaciones como un solo flujo de bits y puede suministrar este flujo de bits único a un CDN que sirve a varios
 40 dispositivos cliente. Los dispositivos, como el dispositivo CDN 16, en el CDN pueden almacenar localmente el flujo de
 bits para su suministro a dispositivos cliente, como el dispositivo de destino 14.

[0062] El dispositivo CDN 16 puede realizar un proceso denominado extracción de sub-flujo de bits a fin de extraer
 sub-flujos de bits de un flujo de bits (es decir, el flujo de bits original). El flujo de bits original puede incluir
 representaciones codificadas de una pluralidad de visualizaciones. Los sub-flujos de bits pueden incluir
 45 representaciones codificadas de un subconjunto de las visualizaciones del flujo de bits original. En algunos ejemplos,
 el dispositivo CDN 16 puede extraer un sub-flujo de bits del flujo de bits original extrayendo selectivamente unidades
 NAL particulares del flujo de bits original. Las unidades NAL extraídas forman el sub-flujo de bits.

50 **[0063]** El dispositivo CDN 16 puede extraer diferentes sub-flujos de bits basándose en las capacidades de diferentes
 dispositivos cliente y/o anchos de banda de transmisión asociados con diferentes dispositivos cliente. Por ejemplo,
 una o más de las múltiples visualizaciones en el flujo de bits original pueden designarse para dispositivos cliente que

5 tienen pantallas más pequeñas, como teléfonos móviles (donde esta visualización puede ser la visualización central descrita anteriormente, que comúnmente es la única visualización requerida para visualizar contenido 3D en una pantalla más pequeña que normalmente es visualizada por una sola persona). En respuesta a la determinación de que el dispositivo cliente que solicita el flujo de bits original es un teléfono móvil, el dispositivo CDN 16 puede extraer un sub-flujo de bits particular del flujo de bits original y suministrar el flujo de bits particular al teléfono móvil. En este ejemplo, el sub-flujo de bits particular puede incluir una representación codificada solo para una de las visualizaciones en el flujo de bits original.

10 **[0064]** Para 3DV compatible con MVC, el proceso de extracción de flujo de bits puede ser descrito de la siguiente manera con respecto al documento anteriormente mencionado N12544. La referencia a las subcláusulas siguiente se refiere a las subcláusulas del documento N12544 u otros documentos relacionados con la extensión 3DV de H.264/AVC, a los que se puede hacer referencia en el documento N12544. A continuación se reproducen o se mencionan varias subcláusulas del documento N12544 y se describe el proceso de extracción del flujo de bits. El proceso de extracción de sub-flujo de bits se describe de la siguiente manera.

15 **[0065]** Especificación de subconjuntos de flujo de bits: Se aplican las especificaciones de la subcláusula H.8.5 del anexo H de H.264.

20 **[0066]** Lo siguiente se aplica al próximo Anexo I de H.264.

25 **[0067]** I.8.5.1 Proceso de obtención para componentes de visualización de anclaje requeridos: La especificación de la subcláusula H.8.5.1 se aplica con la sustitución de "componente de visualización" por "componente de visualización de profundidad" o "componente de visualización de textura" y "visualización" por "visualización de profundidad" o "visualización de textura".

30 **[0068]** I.8.5.2 Proceso de obtención para componentes de visualización sin anclaje requeridos: La especificación de la subcláusula H.8.5.2 se aplica con la sustitución de "componente de visualización" por "componente de visualización de profundidad" o "componente de visualización de textura" y "visualización" por "visualización de profundidad" o "visualización de textura".

35 **[0069]** I.8.5.3 Proceso de extracción de sub-flujo de bits: Es un requisito de la conformidad del flujo de bits que cualquier sub-flujo de bits que sea la salida del proceso especificado en esta subcláusula con pldTarget igual a cualquier valor en el rango de 0 a 63, inclusive, tldTarget igual a cualquier valor en el rango de 0 a 7, inclusive, viewldTargetList consistente en uno o más valores de viewldTarget que identifican las visualizaciones en el flujo de bits, se ajustará a esta Recomendación | Estándar internacional. NOTA 1 - Un flujo de bits conforme contiene una o más unidades NAL de fragmento codificado con priority_id igual a 0 y temporal_id igual a 0. NOTA 2 - También es posible que no todos los puntos de operación de los sub-flujos de bits resultantes del proceso de extracción de sub-flujo de bits tengan un level_idc o level_idc[i] aplicables. En este caso, cada secuencia de vídeo codificada en un flujo de bits puede cumplir con uno o más de los perfiles especificados en el Anexo A, Anexo H y Anexo I, en ITU-T H.264, pero puede no cumplir con las restricciones de nivel especificadas en los apartados A.3, H.10.2 e I.10.2, respectivamente.

40 **[0070]** Las entradas a este proceso de extracción de sub-flujo de bits incluyen: 1) una variable depthPresentFlagTarget (cuando está presente), 2) una variable pldTarget (cuando está presente), 3) una variable tldTarget (cuando está presente), y 4) una lista viewldTargetList que consta de uno o más valores de viewldTarget (cuando está presente). Las salidas de este proceso son un sub-flujo de bits y una lista de valores VOIdx VOIdxList. Además, cuando depthPresentFlagTarget no está presente como entrada a esta subcláusula, se infiere que depthPresentFlagTarget es igual a 0. Cuando pldTarget no está presente como entrada a esta subcláusula, se infiere que pldTarget es igual a 63. Cuando tldTarget no está presente como entrada a esta subcláusula, se infiere que es igual a 7. Cuando viewldTargetList no está presente como entrada en esta subcláusula, debe haber un valor de viewldTarget inferido en viewldTargetList y el valor de viewldTarget se infiere que es igual a view_id de la visualización base.

45 **[0071]** El sub-flujo de bits se obtiene mediante la aplicación de las siguientes operaciones en orden secuencial:

- 55 1. Deje que VOIdxList esté vacío y minVOIdx sea el valor VOIdx de la visualización base.
2. Para cada valor de viewldTarget incluido en viewldTargetList, invoque el proceso especificado en la subcláusula H.8.5.1 para visualizaciones de textura con viewldTarget como entrada.
- 60 3. Si depthPresentFlagTarget es igual a 1, para cada valor de viewldTarget incluido en viewldTargetList, invoque el proceso especificado en la subcláusula H.8.5.1 para visualizaciones de profundidad con viewldTarget como entrada.
- 65 4. Para cada valor de viewldTarget incluido en viewldTargetList, invoque el proceso especificado en la subcláusula H.8.5.2 para visualizaciones de textura con el valor de viewldTarget como entrada.

5. Si depthPresentFlagTarget es igual a 1, para cada valor de viewIdTarget incluido en viewIdTargetList, invoque el proceso especificado en la subcláusula H.8.5.2 para visualizaciones de profundidad con viewIdTarget como entrada.
- 5 6. Marque todas las unidades VCL NAL y las unidades NAL de datos de relleno para las cuales cualquiera de las siguientes condiciones son verdaderas como "para eliminar del flujo de bits":
- 10 a) priority_id es mayor que pldTarget,
- b) temporal_id es mayor que tldTarget,
- c) anchor_pic_flag es igual a 1 y view_id no está marcado como "requerido para el anclaje"
- 15 d) anchor_pic_flag es igual a 0 y view_id no está marcado como "requerido para sin anclaje"
- e) nal_ref_idc es igual a 0 e inter_view_flag es igual a 0 y view_id no es igual a ningún valor en la lista viewIdTargetList,
- 20 f) nal_unit_type es igual a 21 y depthPresentFlagTarget es igual a 0.
7. Elimine todas las unidades de acceso para las cuales todas las unidades VCL NAL están marcadas como "para eliminar del flujo de bits".
8. Elimine todas las unidades VCL NAL y las unidades NAL de datos de relleno que están marcadas como "para eliminar del flujo de bits".
- 25 9. Cuando VOIdxList contiene solo un valor de VOIdx que es igual a minVOIdx, elimine las siguientes unidades NAL:
- 30 a) todas las unidades NAL con nal_unit_type igual a 14 o 15,
- b) todas las unidades NAL con nal_unit_type igual a 6 en las que el primer mensaje SEI tiene un tipo de carga útil en el rango de 36 a 44, inclusive.
- 35 Cuando VOIdxList contiene solo un valor de VOIdx igual a minVOIdx, el sub-flujo de bits contiene solo la visualización base o solo un subconjunto temporal de la visualización base.
10. Cuando depthPresentFlagTarget es igual a 0, elimine todas las unidades NAL con nal_unit_type igual a 6 en las que el primer mensaje SEI tenga payloadType en el rango de 45 a 47, inclusive.
- 40 11. Deje que maxTId sea el temporal_id máximo de todas las unidades VCL NAL restantes. Elimine todas las unidades NAL con nal_unit_type igual a 6 que solo contengan mensajes SEI que son parte de un mensaje SEI de agrupamiento escalable MVC con cualquiera de las siguientes propiedades:
- 45 a) operation_point_flag es igual a 0 y todos los componentes de visualización -in-au-flag son iguales a 0 y ninguno de sei_view_id[i] para todos los i en el rango de 0 a num_view_components_minus1, inclusive, corresponde a un valor VOIdx incluido en VOIdxList,
- 50 b) operation_point_flag es igual a 1 y sei_op_temporal_id es mayor que maxTId o la lista de sei_op_view_id[i] para todos los i en el rango de 0 a num_view_components_op_minus1, inclusive, no es un subconjunto de viewIdTargetList (es decir, no es cierto que sei_op_view_id[i] para cualquier i en el rango de 0 a num_view_components_op_minus1, inclusive, es igual a un valor en viewIdTargetList).
- 55 12. Elimine cada mensaje SEI de información de escalabilidad de visualización y cada mensaje SEI no presente en punto de operación, cuando esté presente.
- 60 13. Cuando VOIdxList no contiene un valor de VOIdx igual a minVOIdx, la visualización con VOIdx igual al valor mínimo de VOIdx incluido en VOIdxList se convierte en la visualización base del sub-flujo de bits extraído. Un procedimiento informativo que describe los pasos clave del proceso para crear una visualización base se describe en la subcláusula I.8.5.6. Cuando VOIdxList no contiene un valor de VOIdx igual a minVOIdx, el sub-flujo de bits resultante de acuerdo con los pasos de operación 1-9 anteriores no contiene una visualización base que se ajuste a uno o más perfiles especificados en el Anexo A. En este caso, en este paso de operación, la visualización restante con el nuevo valor VOIdx mínimo se convierte en la nueva visualización base que se ajusta a uno o más perfiles especificados en el Anexo A y el Anexo H.
- 65

[0072] En la realización de este proceso de extracción de sub-flujo de bits, el dispositivo CDN 16 puede reducir los requisitos de ancho de banda de salida del CDN a un dispositivo cliente, tal como el dispositivo de destino 14. Para facilitar esta reducción del ancho de banda, 3DV compatible con MVC presenta una serie de técnicas de codificación de vídeo que facilitan el proceso de extracción de sub-flujo de bits. Por ejemplo, 3DV compatible con MVC proporciona predicción inter no solo dentro de una visualización sino a través de visualizaciones. La predicción entre visualizaciones en general se permite en 3DV compatible con MVC entre imágenes en la misma unidad de acceso, pero entre imágenes en diferentes unidades de acceso. A este respecto, la descodificación de un componente de visualización particular en una unidad de acceso dada puede requerir la descodificación de uno o más componentes de visualización diferentes en la unidad de acceso dada u otras unidades de acceso. En otras palabras, el componente de visualización particular puede depender de uno o más componentes de visualización diferentes en la unidad de acceso dada u otras unidades de acceso. Las estructuras 3DV compatibles con MVC acceden a unidades y conjuntos de parámetros (por ejemplo, conjuntos de parámetros de secuencia, conjuntos de parámetros de imagen, etc.) de modo que los dispositivos CDN, como el dispositivo CDN 16, puedan determinar las dependencias entre componentes de visualización sin tener que descodificar ninguna visualización de componentes. Por el contrario, los dispositivos CDN pueden determinar las dependencias entre los componentes de visualización basándose en listas de visualizaciones de objetivos señaladas en conjuntos de parámetros de secuencia.

[0073] Las dependencias entre componentes de visualización (que también pueden denominarse "relaciones de predicción") pueden ser diferentes para imágenes de anclaje y sin anclaje. Una imagen de anclaje es una imagen codificada en la que todos los fragmentos de la imagen codificada hacen referencia solo a los fragmentos dentro de la misma unidad de acceso. Consecuentemente, la predicción entre visualizaciones se puede usar en una imagen de anclaje, pero no se usa predicción inter (es decir, la predicción de la unidad de acceso inter) en la imagen de anclaje. Todas las imágenes codificadas que siguen una imagen de anclaje en orden de salida no usan predicción inter de ninguna imagen antes de la imagen codificada en orden de descodificación. Una imagen sin anclaje se refiere a cualquier imagen distinta a una imagen de anclaje.

[0074] Típicamente, las imágenes de anclaje se utilizan de manera periódica dentro del flujo de bits para permitir la descodificación oportuna del contenido. Por ejemplo, el codificador de vídeo 20 puede insertar periódicamente una imagen de anclaje en el flujo de bits para que un descodificador de vídeo, tal como un descodificador de vídeo 30, pueda descodificar el uno o más componentes de visualización sin tener que almacenar en memoria intermedia una cantidad significativa de imágenes adicionales. Por esta razón, las imágenes de anclaje pueden facilitar los cambios de canal reduciendo los tiempos de descodificación a un límite de tiempo máximo conocido tolerable por un consumidor de dicho contenido.

[0075] Para soportar el proceso de extracción de sub-flujo de bits, 3DV compatible con MVC proporciona diferentes reglas para formar sub-flujos de bits con respecto a imágenes de anclaje y sin anclaje sin distinguir entre los diferentes componentes de visualización de cualquiera de las imágenes de anclaje y sin anclaje. Es decir, una visualización codificada en general incluye no solo un componente de visualización de textura sino también un componente de visualización de profundidad para que se pueda realizar vídeo en 3D. En algunos casos, el componente de visualización de profundidad requiere que el componente de visualización de textura se descodifique apropiadamente, mientras que en otros casos, el componente de visualización de profundidad no requiere que el componente de visualización de textura se descodifique apropiadamente. Por lo tanto, el 3DV compatible con MVC como se propone actualmente puede extraer ciertos componentes de visualización de textura del flujo de bits cuando se forma el sub-flujo de bits a pesar de que tales componentes de visualización de textura pueden no ser necesarios para descodificar el componente de visualización de profundidad. En otras palabras, para cada conjunto de valores de view_id seleccionados para salida, una vez que se identifica una visualización de profundidad o de textura para ser utilizada por una imagen de anclaje o una imagen sin anclaje para predicción entre visualizaciones, todas las unidades VCL NAL de esa visualización, independientemente de si pertenecen a un componente de visualización de textura o un componente de visualización de profundidad, se incluyen en el sub-flujo de bits extraído. La no extracción de los componentes innecesarios de la visualización de textura del flujo de bits (es decir, incluidos los componentes innecesarios de la visualización de textura en el sub-flujo de bits) puede dar como resultado ineficiencias en el ancho de banda.

[0076] De acuerdo con las técnicas de esta divulgación, el dispositivo CDN 16 puede mantener listas de visualizaciones de objetivos independientes para el componente de visualización de textura y el componente de visualización de la profundidad en lugar de mantener una única lista para una imagen codificada, independientemente de su componente de visualización de profundidad y componentes de visualización de textura. Manteniendo estas listas por separado, el dispositivo CDN 16 puede identificar cuándo extraer un componente de visualización de textura y un componente de visualización de profundidad y no siempre envía ambos cuando solo se requiere uno u otro. Este cambio en la forma en que se mantienen las listas de visualizaciones de objetivos puede aplicarse tanto a las listas de visualizaciones de objetivos de anclaje como a las listas de visualizaciones de objetivos sin anclaje.

[0077] Para extraer un sub-flujo de bits de un flujo de bits 3DV, el dispositivo CDN 16 puede determinar una lista de visualizaciones de objetivos de textura para un componente de visualización de textura de una visualización en el flujo de bits 3DV. La lista de visualizaciones de objetivos de textura puede indicar una o más partes del flujo de bits 3DV utilizado para predecir el componente de visualización de textura. Además, el dispositivo CDN 16 puede determinar

una lista de visualizaciones de objetivos de profundidad para un componente de visualización de profundidad de la unidad NAL de la visualización en el flujo de bits 3DV. Esta lista de visualizaciones de objetivos de profundidad puede indicar una o más partes del flujo de bits 3DV utilizado para predecir el componente de visualización de profundidad. El dispositivo CDN 16 puede determinar el sub-flujo de bits basándose al menos en parte en la lista de visualizaciones de objetivos de textura y la lista de visualizaciones de objetivos de profundidad. Debido a que la lista de visualizaciones de objetivos de textura puede incluir identificadores de visualización diferentes a la lista de visualizaciones de objetivos de profundidad, el dispositivo CDN 16 puede extraer un componente de visualización de textura sin extraer el componente de visualización de profundidad asociado. Asimismo, el dispositivo CDN 16 puede extraer un componente de visualización de profundidad sin extraer el componente de visualización de textura asociado. Un componente de visualización de textura y un componente de visualización de profundidad se pueden considerar asociados si corresponden a la misma ubicación de cámara (es decir, la misma visualización).

[0078] Como se indica anteriormente, un SPS puede especificar posibles dependencias entre visualizaciones y componentes de visualización aplicables a la SPS. El dispositivo CDN 16 puede determinar, basándose al menos en parte en la pluralidad de visualizaciones de objetivos y las posibles dependencias especificadas por el SPS, la lista de visualizaciones de objetivos de textura. Asimismo, el dispositivo CDN 16 puede determinar, basándose al menos en parte en la pluralidad de visualizaciones de objetivos y las posibles dependencias especificadas por el SPS, la lista de visualizaciones de objetivos de profundidad.

[0079] En algunos casos, las cabeceras de unidades NAL en el flujo de bits pueden incluir elementos sintácticos `depth_to_view`, como se describe anteriormente. De acuerdo con las técnicas de esta divulgación, la semántica del elemento sintáctico `depth_to_view_flag` se puede extender para indicar si un componente de visualización de profundidad se puede descodificar con éxito sin descodificar el componente de visualización de textura asociado. Es decir, el elemento sintáctico `depth_to_view_flag` de una unidad NAL puede indicar si un fragmento codificado de un componente de visualización de profundidad encapsulado por la unidad NAL hace referencia a un componente de visualización de textura. De esta manera, incluso cuando el componente de visualización de textura no está en la lista de visualizaciones de objetivos a extraer para textura, si el elemento sintáctico `depth_to_view_flag` de una unidad NAL está establecido en 1 y el componente de visualización de profundidad correspondiente está en la lista de visualizaciones de objetivos para extraer para profundidad, se pueden extraer las unidades VCL NAL del componente de visualización de textura.

[0080] De forma alternativa, un elemento sintáctico (`use_texture_flag`) de indicador de textura de uso se puede introducir en las cabeceras de unidad NAL. El elemento sintáctico de indicador de textura de uso puede indicar si un componente de visualización de profundidad puede descodificarse con éxito sin descodificar el componente de visualización de textura asociado. En otras palabras, el elemento sintáctico de indicador de textura de uso de una unidad NAL puede indicar si un componente de visualización de profundidad encapsulado por la unidad NAL es descodificable sin descodificar un componente de visualización de textura que corresponde al componente de visualización de profundidad. El elemento sintáctico de indicador de textura de uso puede establecerse en 0 para componentes de visualización de textura y puede establecerse en 1 para componentes de visualización de profundidad cuando el componente de visualización de profundidad requiere el componente de visualización de textura correspondiente para una descodificación correcta. El elemento sintáctico de indicador de textura de uso puede establecerse en 0 para un componente de visualización de profundidad cuando el componente de visualización de profundidad no es necesario para el componente de visualización de textura correspondiente para una descodificación correcta.

[0081] El uso de estos indicadores puede facilitar, en el dispositivo CDN 16, la realización más eficiente del proceso de extracción de sub-flujo de bits cuando se están utilizando tanto la lista de visualizaciones de objetivos de textura como las listas de visualizaciones de objetivos de profundidad.

[0082] 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. La FIG. 2 se proporciona con fines explicativos y no debería considerarse limitadora de las técnicas tales como las ampliamente ejemplificadas y descritas en esta divulgación. Con fines explicativos, esta divulgación describe el codificador de vídeo 20 en el contexto de la codificación H.264/AVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otros estándares o procedimientos de codificación.

[0083] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una unidad de procesamiento de predicción 100, una unidad de generación residual 102, una unidad de procesamiento de transformación 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de procesamiento de transformación inversa 110, una unidad de reconstrucción 112, una unidad de filtro 113, una memoria intermedia de imágenes descodificadas 114, y una unidad de codificación por entropía 116. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de predicción inter 121 y una unidad de procesamiento de predicción intra 126. La unidad de procesamiento de predicción inter 121 incluye una unidad de estimación de movimiento 122 y una unidad de compensación de movimiento 124. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0084] El codificador de vídeo 20 recibe datos de vídeo. Para codificar los datos de vídeo, el codificador de vídeo 20 puede codificar cada fragmento de cada imagen de los datos de vídeo. Como parte de la codificación de un fragmento, el codificador de vídeo 20 puede codificar unidades de vídeo del fragmento.

5 **[0085]** La unidad de procesamiento de predicción inter 121 puede generar datos predictivos para una unidad de vídeo actual realizando una predicción inter. Los datos predictivos para la unidad de vídeo actual pueden incluir bloques predictivos e información de movimiento para la unidad de vídeo actual.

10 **[0086]** Los fragmentos pueden ser fragmentos I, fragmentos P, fragmentos SP, o fragmentos B. La unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 pueden realizar diferentes operaciones para una unidad de vídeo dependiendo de si la unidad de vídeo está en un fragmento I, un fragmento P, un fragmento SP o un fragmento B. En un fragmento I, todas las unidades de vídeo están predecibles. Por lo tanto, si la unidad de vídeo está en un fragmento I, la unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 no realizan una predicción inter en la unidad de vídeo. Un fragmento SP es un fragmento que puede codificarse mediante predicción intra o predicción inter con cuantificación de las muestras de predicción usando como máximo un vector de movimiento e índice de referencia para predecir los valores de muestra de cada bloque. Un fragmento SP puede codificarse de modo que sus muestras descodificadas se puedan construir de forma idéntica a otro fragmento SP o un fragmento SI. Un fragmento SI es un fragmento que se codifica mediante predicción intra solo y mediante cuantificación de las muestras de predicción. Un fragmento SI puede codificarse de modo que sus muestras descodificadas se puedan construir de forma idéntica a un fragmento SP.

25 **[0087]** La unidad de procesamiento de predicción inter 121 puede realizar un proceso de construcción de la lista de imágenes de referencia al comienzo de la codificación de cada fragmento P, SP o B. Si la unidad de procesamiento de predicción inter 121 está codificando un fragmento P o SP, la unidad de procesamiento de predicción inter 121 puede generar una primera lista de imágenes de referencia (por ejemplo, lista 0). Si la unidad de procesamiento de predicción inter 121 codifica un fragmento B, la unidad de procesamiento de predicción inter 121 puede generar la primera lista de imágenes de referencia (por ejemplo, lista 0) y también generar una segunda lista de imágenes de referencia (por ejemplo, lista 1).

30 **[0088]** Como se mencionó anteriormente, la extensión de cabecera de unidad de MVC NAL incluye un elemento sintáctico `non_idr_flag`. Si el elemento sintáctico `non_idr_flag` indica que la unidad NAL encapsula un fragmento codificado de una imagen IDR (por ejemplo, el elemento sintáctico `non_idr_flag` es igual a 0), la unidad de procesamiento de predicción inter 121 puede generar la lista 0 (y la lista 1 si el fragmento codificado es un fragmento B codificado) de manera que todas las entradas no indiquen ninguna imagen de referencia. De lo contrario, si el elemento sintáctico `non_idr_flag` indica que la unidad NAL encapsula un fragmento codificado de una imagen que no es IDR, la unidad de procesamiento de predicción inter 121 puede generar una versión inicial de la lista 0 y, para un fragmento B, una versión inicial de la lista 1 como se describe en la sección 8.2.4.1 del estándar H.264/AVC.

40 **[0089]** Además, después de la generación de la versión inicial de la lista 0 y, para un fragmento B, la versión inicial de la lista 1, la unidad de procesamiento de predicción inter 121 puede agregar componentes de referencia entre visualizaciones o componentes solo de referencia entre visualizaciones a la versión inicial de la lista 0 y, para un fragmento B, la versión inicial de la lista 1. Después de agregar los componentes de referencia entre visualizaciones o componentes de referencia entre visualizaciones solamente, la unidad de procesamiento de predicción inter 121 puede realizar un proceso de modificación de lista de imágenes de referencia para generar una versión final de la lista 0 y, para un fragmento B, una versión final de la lista 1.

50 **[0090]** Para agregar los componentes de referencia entre visualizaciones a la lista 0 y la lista 1, la unidad de procesamiento de predicción inter 121 puede primero determinar si el fragmento codificado encapsulado por la unidad NAL (es decir, el fragmento actual) es parte de una imagen de anclaje. Como se describió anteriormente, la extensión de cabecera de la unidad MVC NAL puede incluir el elemento sintáctico `anchor_pic_flag`. El elemento sintáctico `anchor_pic_flag` de la unidad NAL puede indicar si el fragmento actual es parte de una imagen de anclaje. Si el fragmento actual es parte de una imagen de anclaje, la unidad de procesamiento de predicción inter 121 puede agregar a la lista X (donde X es 0 o 1) cada imagen de referencia entre visualizaciones que pertenece a la misma unidad de acceso que el fragmento actual y se especifica para imágenes de anclaje en el SPS aplicable. Por lo tanto, las únicas imágenes de referencia entre visualizaciones en la lista X pertenecen a la misma unidad de acceso que el fragmento actual.

60 **[0091]** Por ejemplo, como se indicó anteriormente, la extensión MVC SPS puede incluir elementos sintácticos `num_anchor_refs_l0[i]`, elementos sintácticos `num_anchor_refs_l1`, elementos sintácticos `anchor_ref_l0[i][j]` y elementos sintácticos `anchor_ref_l1[i][j]`. En este ejemplo, si el fragmento actual tiene `anchor_pic_flag` igual a 1, la unidad de procesamiento de predicción inter 121 puede, para cada valor del índice de visualización de referencia j de 0 a `num_anchor_refs_l0[i] - 1`, inclusive, en orden ascendente de j, agregar a la lista 0 la referencia de predicción entre visualizaciones con id de la visualización igual a `anchor_ref_l0[i][j]` desde la misma unidad de acceso que el fragmento actual. De forma similar, si el fragmento actual tiene `anchor_pic_flag` igual a 1, la unidad de procesamiento de predicción inter 121 puede, para cada valor del índice de visualización de referencia j de 0 a `num_anchor_refs_l1[i] -`

1, inclusive, en orden ascendente de j , agregar a la lista 1 la referencia de predicción entre visualizaciones con id de la visualización igual a $\text{anchor_ref_l1}[i][j]$ desde la misma unidad de acceso que el fragmento actual.

5 **[0092]** De lo contrario, si el fragmento actual no es parte de una imagen de anclaje, la unidad de procesamiento de predicción inter 121 puede agregar a la lista X (donde X es 0 o 1) cada imagen de referencia entre visualizaciones que pertenece a la misma unidad de acceso como el fragmento actual y se especifica para las imágenes sin anclaje en el SPS aplicable. Por lo tanto, las únicas imágenes de referencia entre visualizaciones en la lista X pertenecen a la misma unidad de acceso que el fragmento actual.

10 **[0093]** Por ejemplo, como se indicó anteriormente, la extensión MVC SPS puede incluir elementos sintácticos $\text{num_non_anchor_refs_l0}[i]$, elementos sintácticos $\text{num_non_anchor_refs_l1}$, elementos sintácticos $\text{non_anchor_ref_l0}[i][j]$, y elementos sintácticos $\text{non_anchor_ref_l1}[i][j]$. En este ejemplo, si el fragmento actual tiene anchor_pic_flag igual a 0, la unidad de procesamiento de predicción inter 121 puede, para cada valor del índice de visualización de referencia j de 0 a $\text{num_non_anchor_refs_l0}[i] - 1$, inclusive, en orden ascendente de j , agregar a la lista 0 la referencia de predicción entre visualizaciones con view_id igual a $\text{non_anchor_ref_l0}[i][j]$ desde la misma unidad de acceso que el fragmento actual. De manera similar, si el fragmento actual tiene anchor_pic_flag igual a 0, la unidad de procesamiento de predicción inter 121 puede, para cada valor del índice de visualización de referencia j de 0 a $\text{num_non_anchor_refs_l1}[i] - 1$, inclusive, en orden ascendente de j , agregar a la lista 1 la referencia de predicción entre visualizaciones con id de la visualización igual a $\text{non_anchor_ref_l1}[i][j]$ desde la misma unidad de acceso que el fragmento actual.

25 **[0094]** Si la unidad de vídeo actual está en un fragmento P, la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia de una lista de imágenes de referencia (por ejemplo, lista 0) para un bloque de referencia para la unidad de vídeo actual. El bloque de referencia de la unidad de vídeo puede incluir un bloque de luma y los correspondientes bloques de croma que se corresponden más estrechamente con los bloques de luma y croma de la unidad de vídeo actual. La unidad de estimación de movimiento 122 puede usar una variedad de métricas para determinar con qué exactitud los bloques de referencia en una imagen de referencia corresponden a los bloques de luma y croma de la unidad de vídeo actual. Por ejemplo, la unidad de estimación de movimiento 122 puede determinar con qué exactitud un bloque de referencia en una imagen de referencia corresponde a los bloques de luma y croma de la unidad de vídeo actual mediante una suma de diferencias absolutas (SAD), suma de diferencias cuadradas (SSD) u otra métricas de diferencia.

35 **[0095]** La unidad de estimación de movimiento 122 puede generar un índice de referencia que indica la imagen de referencia de la lista 0 que contiene un bloque de referencia de una unidad de vídeo actual (por ejemplo, una partición MB o MB) en un fragmento P y un vector de movimiento que indica un desplazamiento espacial entre los bloques de la unidad de vídeo actual y el bloque de referencia. La unidad de estimación de movimiento 122 puede emitir el índice de referencia y el vector de movimiento como la información de movimiento de la unidad de vídeo. La unidad de compensación de movimiento 124 puede generar los bloques predictivos de luma y croma para la unidad de vídeo actual basándose en los bloques de luma y croma del bloque de referencia indicado por la información de movimiento de la unidad de vídeo actual.

45 **[0096]** Si la unidad de vídeo actual está en un fragmento B, la unidad de estimación de movimiento 122 puede llevar a cabo predicción inter unidireccional o predicción inter bidireccional para la unidad de vídeo actual. Para realizar una predicción inter unidireccional para la unidad de vídeo actual, la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia de la lista 0 o una segunda lista de imágenes de referencia (por ejemplo, lista 1) para un bloque de referencia para la unidad de vídeo. En ejemplos donde el codificador de vídeo 20 usa MVC o 3DV, la lista 0 y/o la lista 1 pueden incluir imágenes de referencia entre visualizaciones. La unidad de estimación de movimiento 122 puede generar un índice de referencia que indique una posición en la lista 0 o lista 1 de la imagen de referencia que contiene un bloque de referencia. Además, la unidad de estimación de movimiento 122 puede determinar un vector de movimiento que indique un desplazamiento espacial entre los bloques de la unidad de vídeo actual y el bloque de referencia. La unidad de estimación de movimiento 122 también puede generar un indicador de dirección de predicción que indica si la imagen de referencia está en la lista 0 o en la lista 1.

55 **[0097]** Para realizar predicción inter bidireccional para una unidad de vídeo actual (por ejemplo, un MB o partición MB), la unidad de estimación de movimiento 122 puede buscar las imágenes de referencia de la lista 0 para un bloque de referencia y también puede buscar las imágenes de referencia de la lista 1 para otro bloque de referencia. La unidad de estimación de movimiento 122 puede generar índices de referencia que indiquen posiciones en la lista 0 y lista 1 de las imágenes de referencia que contienen los bloques de referencia. Además, la unidad de estimación de movimiento 122 puede determinar vectores de movimiento que indican desplazamientos espaciales entre los bloques de referencia y los bloques de la unidad de vídeo actual. La información de movimiento de la unidad de vídeo actual puede incluir los índices de referencia y los vectores de movimiento de la unidad de vídeo actual. La unidad de compensación de movimiento 124 puede generar bloques predictivos de luma y croma para la unidad de vídeo actual basándose en los bloques de referencia indicados por la información de movimiento de la unidad de vídeo actual.

65 **[0098]** La unidad de procesamiento de predicción intra 126 puede generar datos predictivos para una unidad de vídeo actual mediante la realización de predicción intra. Los datos predictivos para la unidad de vídeo actual pueden incluir

bloques predictivos para la unidad de vídeo actual y varios elementos sintácticos. La unidad de procesamiento de predicción intra 126 puede realizar la predicción intra para unidades de vídeo en fragmentos I, fragmentos P, fragmentos SP y fragmentos B.

5 **[0099]** Para llevar a cabo la predicción intra en una unidad de vídeo actual, la unidad de procesamiento de predicción intra 126 puede utilizar múltiples modos de predicción intra para generar múltiples conjuntos de datos predictivos para la unidad de vídeo actual. Para usar un modo de predicción intra para generar un conjunto de datos predictivos para la unidad de vídeo actual, la unidad de procesamiento de predicción intra 126 puede extender muestras de bloques contiguos a través de los bloques de la unidad de vídeo actual en una dirección asociada con el modo de predicción
10 intra. Los bloques contiguos pueden estar arriba, arriba y a la derecha, arriba y a la izquierda, o a la izquierda de los bloques de la unidad de vídeo actual, suponiendo un orden de codificación de arriba a abajo, de izquierda a derecha para unidades de vídeo. En algunos ejemplos, el número de modos de predicción intra puede depender del tamaño de los bloques de la unidad de vídeo actual. La unidad de procesamiento de predicción intra 126 selecciona uno de los modos de predicción intra para seleccionar los bloques predictivos para la unidad de vídeo actual.

15 **[0100]** La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para una unidad de vídeo actual entre los datos predictivos generados por la unidad de procesamiento de predicción inter 121 para la unidad de vídeo actual o los datos predictivos generados por la unidad de procesamiento de predicción 126 para la unidad de vídeo actual. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos
20 predictivos para la unidad de vídeo actual basándose en las métricas de velocidad/distorsión de los conjuntos de datos predictivos.

25 **[0101]** La unidad de generación residual 102 puede generar bloques residuales de luma y croma restando las muestras de los bloques predictivos de luma y croma de las muestras correspondientes de los bloques de luma y croma de la unidad de vídeo actual. La unidad de procesamiento de transformación 104 puede generar bloques de coeficientes de transformación para cada bloque residual aplicando una o más transformaciones al bloque residual. La unidad de procesamiento de transformación 104 puede aplicar diversas transformaciones a un bloque residual. Por ejemplo, la unidad de procesamiento de transformación 104 puede aplicar una transformación de coseno discreta (DCT), una transformación direccional o una transformación conceptualmente similar al bloque residual.
30

[0102] La unidad de cuantificación 106 puede cuantificar los coeficientes de transformación en un bloque de coeficientes de transformación. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes de transformación. Por ejemplo, un coeficiente de transformación de n bits puede redondearse hacia abajo a un coeficiente de transformación de m bits durante la cuantificación, donde n es mayor que
35 m . La unidad de cuantificación 106 puede cuantificar un bloque de coeficientes de transformación basándose en un valor de parámetro de cuantificación (QP). El codificador de vídeo 20 puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes de transformación ajustando el valor de QP.

40 **[0103]** La unidad de cuantificación inversa 108 y la unidad de procesamiento de transformación inversa 110 pueden aplicar cuantificación inversa y transformaciones inversas a un bloque de coeficientes de transformación, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficientes. La unidad de reconstrucción 112 agrega muestras en bloques residuales reconstruidos a muestras correspondientes de uno o más bloques predictivos generados por la unidad de procesamiento de predicción 100 para producir bloques reconstruidos. La unidad de filtro 113 puede realizar una operación de desbloqueo para reducir las distorsiones de bloqueo en los bloques reconstruidos. La memoria intermedia de imágenes descodificadas 114 puede almacenar los bloques reconstruidos después de que la unidad de filtro 113 realice la una o más operaciones de desbloqueo en los bloques reconstruidos. La unidad de estimación de movimiento 122 y la unidad de compensación de movimiento 124 pueden usar una imagen de referencia que contiene los bloques reconstruidos para realizar la predicción inter para unidades de vídeo de imágenes posteriores. Además, la unidad de procesamiento de predicción intra 126 puede usar bloques reconstruidos en la memoria intermedia de imágenes descodificadas 114 para realizar la predicción intra.
50

[0104] La unidad de codificación por entropía 116 puede recibir datos de otros componentes funcionales del codificador de vídeo 20. Por ejemplo, la unidad de codificación por entropía 116 puede recibir bloques de coeficientes de la unidad de cuantificación 106 y puede recibir elementos sintácticos de la unidad de procesamiento de predicción 100. La unidad de codificación por entropía 116 puede realizar una o más operaciones de codificación por entropía en los datos para generar datos codificados por entropía. Por ejemplo, el codificador de vídeo 20 puede realizar una operación CAVLC, una operación CABAC, una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética binaria adaptable al contexto basada en sintaxis (SBAC), una operación de codificación por entropía de partición de intervalo de probabilidad (PIPE), una operación de codificación Exponential-Golomb u otro tipo de operación de codificación por entropía en los datos.
60

[0105] 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. La FIG. 3 se proporciona con fines explicativos y no se limita a las técnicas como las ampliamente ejemplificadas y descritas en esta divulgación. Con fines explicativos, esta divulgación describe el descodificador de vídeo 30 en el contexto de la codificación H.264/AVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otros estándares o procedimientos de codificación.
65

[0106] En el ejemplo de la FIG. 3, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 150, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de procesamiento de transformación inversa 156, una unidad de reconstrucción 158, una unidad de filtro 159 y una memoria intermedia de imágenes descodificadas 160. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 162 y la unidad de procesamiento de predicción intra 164. En otros ejemplos, el descodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

[0107] El descodificador de vídeo 30 recibe un flujo de bits. La unidad de descodificación por entropía 150 puede analizar el flujo de bits para extraer elementos sintácticos del flujo de bits. Como parte del análisis del flujo de bits, la unidad de descodificación por entropía 150 puede descodificar por entropía de elementos sintácticos codificados por entropía en el flujo de bits. La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de procesamiento de transformación inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 159 pueden generar datos de vídeo descodificados (es decir, reconstruir los datos de vídeo) basándose en los elementos sintácticos extraídos del flujo de bits. Los elementos sintácticos extraídos del flujo de bits pueden incluir elementos sintácticos que representan bloques de coeficientes de transformación.

[0108] La unidad de cuantificación inversa 154 puede cuantificar inversamente, es decir, descuantificar, transformar bloques de coeficientes. La unidad de cuantificación inversa 154 puede usar un valor de QP para determinar un grado de cuantificación y, del mismo modo, un grado de cuantificación inversa para la aplicación de la unidad de cuantificación inversa 154. Después de que la unidad de cuantificación inversa 154 cuantifica inversamente un bloque de coeficientes de transformación, la unidad de procesamiento de transformación inversa 156 puede aplicar una o más transformaciones inversas al bloque de coeficientes de transformación para generar un bloque residual. Por ejemplo, la unidad de procesamiento de transformación inversa 156 puede aplicar una DCT inversa, una transformación entera inversa, una transformación de Karhunen-Loeve (KLT) inversa, una transformación de rotación inversa, una transformación direccional inversa u otra transformación inversa al bloque de coeficientes de transformación.

[0109] Si una unidad de vídeo actual (por ejemplo, una partición de MB o MB) se codifica utilizando la predicción intra, la unidad de procesamiento de la predicción intra 164 puede realizar la predicción intra para generar bloques predictivos de luma y croma para la unidad de vídeo actual. Por ejemplo, la unidad de procesamiento de predicción intra 164 puede determinar un modo de predicción intra para la unidad de vídeo actual basada al menos en parte en elementos sintácticos en el flujo de bits. La unidad de procesamiento de predicción intra 164 puede usar el modo de predicción intra para generar los bloques predictivos para la unidad de vídeo actual basándose en bloques contiguos espacialmente.

[0110] La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (lista 0) y una segunda lista de imágenes de referencia (lista 1) basándose en los elementos sintácticos extraídos del flujo de bits. En los ejemplos en los que el flujo de bits está codificado utilizando 3DV compatible con MVC o MVC, la lista 0 y/o la lista 1 pueden incluir imágenes de referencia entre visualizaciones. La unidad de procesamiento de predicción 152 puede construir las listas de imágenes de referencia de la misma manera que la descrita anteriormente con referencia a la unidad de procesamiento de predicción inter 121 de la FIG. 2.

[0111] Además, si una unidad de vídeo actual (por ejemplo, una partición de MB o MB) se codifica utilizando la predicción inter, la unidad de descodificación por entropía 150 puede extraer, del flujo de bits, información de movimiento para la unidad de vídeo actual. La unidad de compensación de movimiento 162 puede determinar, basándose al menos en parte en la información de movimiento de la unidad de vídeo actual, uno o más bloques de referencia para la unidad de vídeo actual. La unidad de compensación de movimiento 162 puede generar, basándose al menos en parte en uno o más bloques de referencia para la unidad de vídeo actual, bloques predictivos para la unidad de vídeo actual.

[0112] La unidad de reconstrucción 158 puede reconstruirse, basándose al menos en parte en los bloques residuales de luma y croma para la unidad de vídeo actual y los bloques predictivos de luma y croma de la unidad de vídeo actual, bloques de luma y croma para la unidad de vídeo actual. En particular, la unidad de reconstrucción 158 puede añadir muestras (por ejemplo, componentes de luma o croma) de los bloques residuales a las muestras correspondientes de los bloques predictivos para reconstruir los bloques de luma y croma de la unidad de vídeo actual. La unidad de filtro 159 puede realizar una operación de desbloqueo para reducir las distorsiones de bloque asociadas con los bloques reconstruidos de la unidad de vídeo actual. El descodificador de vídeo 30 puede almacenar los bloques reconstruidos en la memoria intermedia de imágenes descodificadas 160. La memoria intermedia de imágenes descodificadas 160 puede proporcionar imágenes de referencia para la posterior compensación de movimiento, la predicción intra y la presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Por ejemplo, el descodificador de vídeo 30 puede realizar, basándose en los bloques de la memoria intermedia de imágenes descodificadas 160, operaciones de predicción intra o de predicción inter en las PU de otras CU.

[0113] La FIG. 4 es un diagrama de flujo que ilustra un ejemplo de operación de extracción de sub-flujo de bits 200, de acuerdo con una o más técnicas de esta divulgación. El diagrama de flujo de la FIG. 4 y los diagramas de flujo de

las siguientes figuras se proporcionan como ejemplos. En otros ejemplos, las técnicas de esta divulgación se pueden implementar usando más, menos, o diferentes pasos que los mostrados en el ejemplo de la FIG. 4 y las siguientes figuras.

- 5 **[0114]** Como se ilustra en el ejemplo de la FIG. 4, el dispositivo CDN 16 puede determinar una lista de visualizaciones de objetivos de textura que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos (202). Las visualizaciones de objetivos pueden ser un subconjunto de las visualizaciones en el flujo de bits 3DV que se pueden descodificar desde el sub-flujo de bits. Además, el dispositivo CDN 16 puede determinar una lista de visualizaciones de objetivos de profundidad que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos (204). El dispositivo CDN 16 puede determinar el sub-flujo de bits basándose, al menos en parte, en la lista de visualizaciones de objetivos de textura y la lista de visualizaciones de objetivos de profundidad (206).
- 10
- 15 **[0115]** En algunos ejemplos, la lista de visualizaciones de objetivos de textura incluye identificadores de visualización que identifican las visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. Además, la lista de visualizaciones de objetivos de profundidad puede incluir identificadores de visualización que identifiquen las visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos. La lista de visualizaciones de objetivos de textura puede incluir uno o más identificadores de visualización que son diferentes a los identificadores de visualización en la lista de visualizaciones de objetivos de profundidad.
- 20
- 25 **[0116]** Como se describió anteriormente, el flujo de bits 3DV puede comprender una serie de unidades NAL. En algunos ejemplos, el dispositivo CDN 16 puede determinar, basándose al menos en parte en si la lista de visualizaciones de objetivos de textura o la lista de visualizaciones de objetivos de profundidad especifica un identificador de visualización de una unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.
- 30 **[0117]** La FIG. 5 es un diagrama de flujo que ilustra un ejemplo de proceso de extracción de sub-flujo de bits 298 en 3DV compatible con MVC, de acuerdo con una o más técnicas de esta divulgación. El proceso de extracción de sub-flujo de bits 298 puede ser un ejemplo más específico del proceso de extracción de sub-flujo de bits 200 de la FIG. 4. En el ejemplo de la FIG. 5, la lista de visualizaciones de objetivos para los componentes de visualización de textura (por ejemplo, la lista de visualizaciones de objetivos de textura) está dividida en dos partes: una lista de visualizaciones requeridas para la textura de anclaje y una lista de visualizaciones requeridas para una textura sin anclaje. De manera similar, en el ejemplo de la FIG. 5, la lista de visualizaciones de objetivos para los componentes de visualización de profundidad (por ejemplo, la lista de visualizaciones de objetivos de profundidad) está dividida en dos partes: una lista de visualizaciones requeridas para profundidad de anclaje y una lista de visualizaciones requeridas para profundidad sin anclaje.
- 35
- 40 **[0118]** El proceso de extracción de sub-flujo de bits 298 puede tomar varias variables como entradas. Estas entradas pueden incluir un objetivo de indicador presente de profundidad (`depthPresentFlagTarget`), una variable `pldTarget`, una variable `tldTarget` y una lista de objetivos de identificador de visualización (`viewIdTargetList`). La lista de objetivos de identificador de visualización puede consistir en uno o más valores de `viewIdTarget`. Los valores de `viewIdTarget` pueden incluir los identificadores de visualización de las visualizaciones que se incluirán en el sub-flujo de bits objetivo (es decir, el sub-flujo de bits extraído mediante el proceso de extracción de sub-flujo de bits 298). El `depthPresentFlagTarget` puede indicar si las visualizaciones de profundidad están incluidas en el sub-flujo de bits objetivo. Cuando `depthPresentFlagTarget` no está presente como entrada, el dispositivo CDN 16 puede inferir (es decir, determinar automáticamente) que `depthPresentFlagTarget` es igual a 0. `pldTarget` puede identificar un nivel de prioridad objetivo. Cuando `pldTarget` no está presente como entrada, el dispositivo CDN 16 puede inferir que `pldTarget` es igual a 63. `tldTarget` puede identificar un nivel temporal objetivo. Cuando `tldTarget` no está presente como entrada, el dispositivo CDN 16 puede inferir que `tldTarget` es igual a 7. Cuando `viewIdTargetList` no está presente como entrada, el dispositivo CDN 16 puede inferir que hay un valor de `viewIdTarget` en `viewIdTargetList` y que el valor de `viewIdTarget` es igual al identificador de visualización (`view id`) de la visualización base.
- 45
- 50
- 55 **[0119]** Las salidas de proceso de extracción de sub-flujo de bits 298 pueden incluir un sub-flujo de bits y una lista de valores de índice de orden de visualización (`VOIdxList`). En algunos ejemplos, puede ser un requisito de la conformidad del flujo de bits que cualquier sub-flujo de bits que sea el resultado del proceso de extracción de sub-flujo de bits 298 con `pldTarget` sea igual a cualquier valor en el rango de 0 a 63, inclusive, `tldTarget` sea igual a cualquier valor en el rango de 0 a 7, inclusive, donde `viewIdTargetList` consiste en uno o más valores de `viewIdTarget` que identifica la visualización en el flujo de bits conforme al estándar H.264/AVC. Un flujo de bits conforme contiene una o más unidades NAL de fragmentos codificados con `priority_id` igual a 0 y `temporal_id` igual a 0. Es posible que no todos los puntos de operación de un sub-flujo de bits resultante del proceso de extracción de sub-flujo de bits 298 tengan un `level_idc` o `level_idc[i]` aplicable. En este caso, cada secuencia de vídeo codificada en un sub-flujo de bits puede (o debe) seguir cumpliendo con uno o más de los perfiles especificados en el Anexo A, Anexo H y Anexo I del estándar H.264/AVC, pero puede no satisfacer las restricciones de nivel especificadas en los apartados A.3, H.1.0.2 e I.1.0.2, respectivamente, del estándar H.264/AVC.
- 60
- 65

[0120] En el ejemplo de la FIG. 5, el dispositivo CDN 16 puede generar una lista de índices de orden de visualización vacía (VOIdxList) y puede inicializar un índice de orden de visualización mínimo (minVOIdx) para ser el índice de orden de visualización de una visualización base (300). Es decir, el dispositivo CDN 16 puede dejar que VOIdxList esté vacío y puede dejar que minVOIdx sea el valor VOIdx de la visualización base.

[0121] Además, el dispositivo CDN 16 puede generar una lista de visualizaciones de objetivos de textura de anclaje (302). La lista de visualizaciones de objetivos de textura de anclaje puede ser una lista de visualizaciones que tienen componentes de visualización de textura requeridos para descodificar imágenes de anclaje de las visualizaciones de objetivos. Esta divulgación puede referirse a las visualizaciones en la lista de visualizaciones de objetivos de textura de anclaje como visualizaciones marcadas como "requeridas para la textura de anclaje". Cuando el dispositivo CDN 16 genera la lista de visualizaciones de objetivos de textura de anclaje, el dispositivo CDN 16 puede incluir en la lista de índices de orden de visualización (VOIdxList) los índices de orden de visualización de las visualizaciones marcadas como "requeridas para textura de anclaje". Como se describe en otra parte en esta divulgación, el dispositivo CDN 16 puede determinar, basándose al menos en parte en la lista de visualizaciones de objetivos de textura de anclaje, unidades VCL NAL y unidades NAL de datos de relleno para eliminar del flujo de bits. En diversos ejemplos, el dispositivo CDN 16 puede generar la lista de visualizaciones de objetivos de textura de anclaje de diversas maneras. Por ejemplo, el dispositivo CDN 16 puede, para cada valor de viewldTarget incluido en viewldTargetList, invocar un proceso de obtención de la lista de visualizaciones de objetivos de textura de anclaje con viewldTarget como entrada. La FIG. 7, que se describe en detalle más adelante en esta divulgación, es un diagrama de flujo de un ejemplo de proceso de obtención de la lista de visualizaciones de objetivos de textura de anclaje.

[0122] Después de generar la lista de visualizaciones de objetivos de textura de anclaje, el dispositivo CDN 16 puede determinar si los componentes de visualización de profundidad deben incluirse en el sub-flujo de bits (304). En el ejemplo de la FIG. 5, el dispositivo CDN 16 puede determinar, basándose en depthPresentFlagTarget, si los componentes de visualización de profundidad se incluirán en el sub-flujo de bits. En respuesta a la determinación de que los componentes de visualización de profundidad deben incluirse en el sub-flujo de bits ("SÍ" de 304), el dispositivo CDN 16 puede generar una lista de visualizaciones de objetivos de profundidad de anclaje (306). La lista de visualizaciones de objetivos de profundidad de anclaje puede ser una lista de visualizaciones que tienen componentes de visualización de profundidad requeridos para descodificar imágenes de anclaje de las visualizaciones de objetivos. Esta divulgación puede referirse a las visualizaciones en la lista de visualizaciones de objetivos de profundidad de anclaje como visualizaciones que están marcadas como "requeridas para la profundidad de anclaje". Cuando el dispositivo CDN 16 genera la lista de visualizaciones de objetivos de profundidad de anclaje, el dispositivo CDN 16 puede incluir en la lista de índices de orden de visualización (VOIdxList) los índices de orden de visualización de las visualizaciones marcadas como "requeridas para la profundidad de anclaje". Como se describe en otra parte en esta divulgación, el dispositivo CDN 16 puede determinar, basándose al menos en parte en la lista de visualizaciones de objetivos de profundidad de anclaje, unidades VCL NAL y unidades NAL de datos de relleno para eliminar del flujo de bits. En diversos ejemplos, el dispositivo CDN 16 puede generar la lista de visualizaciones de objetivos de profundidad de anclaje de diversas maneras. Por ejemplo, si depthPresentFlagTarget es igual a 1, el dispositivo CDN 16 puede, para cada viewldTarget en viewldTargetList, realizar un proceso de obtención de la lista de visualizaciones de objetivos de profundidad de anclaje con viewldTarget como entrada. La FIG. 8, que se describe en detalle más adelante en esta divulgación, es un diagrama de flujo de un ejemplo de proceso de obtención de la lista de visualizaciones de objetivos de profundidad de anclaje.

[0123] Después de generar la lista de visualizaciones de objetivos de profundidad de anclaje o en respuesta a determinar que los componentes de visualización de profundidad no deben incluirse en el sub-flujo de bits ("NO" de 304), el dispositivo CDN 16 puede generar una lista de visualizaciones de objetivos de textura sin anclaje (308). La lista de visualizaciones de objetivos de textura sin anclaje puede ser una lista de visualizaciones que tienen componentes de visualización de textura requeridos para descodificar imágenes sin anclaje de las visualizaciones de objetivos. Esta divulgación puede referirse a las visualizaciones en la lista de visualizaciones de objetivos de textura sin anclaje como visualizaciones que están marcadas como "requeridas para texturas sin anclaje". Cuando el dispositivo CDN 16 genera la lista de visualizaciones de objetivos de textura sin anclaje, el dispositivo CDN 16 puede incluir en la lista de índices de orden de visualización (VOIdxList) los índices de orden de visualización de las visualizaciones marcadas como "requeridas para textura sin anclaje". Como se describe en otra parte en esta divulgación, el dispositivo CDN 16 puede determinar, basándose al menos en parte en la lista de visualizaciones de objetivos de textura sin anclaje, unidades VCL NAL y unidades NAL de datos de relleno para eliminar del flujo de bits. En varios ejemplos, el dispositivo CDN 16 puede generar la lista de visualizaciones de objetivos de textura sin anclaje de varias maneras. Por ejemplo, el dispositivo CDN 16 puede, para cada valor de viewldTarget incluido en viewldTargetList, invocar un proceso de obtención de la lista de visualizaciones de objetivos de textura sin anclaje con el viewldTarget como entrada. La FIG. 9, que se describe en detalle más adelante en esta divulgación, es un diagrama de flujo de un ejemplo de proceso de obtención de la lista de visualizaciones de objetivos de textura sin anclaje.

[0124] Además, el dispositivo CDN 16 puede determinar si los componentes de visualización de profundidad deben incluirse en el sub-flujo de bits (310). En respuesta a la determinación de que los componentes de visualización de profundidad deben incluirse en el sub-flujo de bits ("SÍ" de 310), el dispositivo CDN 16 puede generar una lista de visualizaciones de objetivos de profundidad sin anclaje (312). La lista de visualizaciones de objetivos de profundidad

sin anclaje puede ser una lista de visualizaciones que tienen componentes de visualización de profundidad requeridos para descodificar imágenes sin anclaje de las visualizaciones de objetivos. Esta divulgación puede referirse a las visualizaciones en la lista de visualizaciones de objetivos de profundidad sin anclaje como visualizaciones que están marcadas como "requeridas para profundidad sin anclaje". Cuando el dispositivo CDN 16 genera la lista de visualizaciones de objetivos de profundidad sin anclaje, el dispositivo CDN 16 puede incluir en la lista de índices de orden de visualización (VOIdxList) los índices de orden de visualización de las visualizaciones que están marcadas como "requeridas para profundidad sin anclaje". Como se describe en otra parte en esta divulgación, el dispositivo CDN 16 puede determinar, basándose al menos en parte en la lista de visualizaciones de objetivos de profundidad sin anclaje, unidades VCL NAL y unidades NAL de datos de relleno para eliminar del flujo de bits. En diversos ejemplos, el dispositivo CDN 16 puede generar la lista de visualizaciones de objetivos de profundidad sin anclaje de varias maneras. Por ejemplo, si depthPresentFlagTarget es igual a 1, el dispositivo CDN 16 puede, para cada valor de viewIdTarget incluido en viewIdTargetList, invocar un proceso de obtención de la lista de visualizaciones de objetivos de profundidad sin anclaje con viewIdTarget como entrada. La FIG. 10, que se describe en detalle más adelante en esta divulgación, es un diagrama de flujo de un ejemplo de proceso de obtención de la lista de visualizaciones de objetivos de textura sin anclaje.

[0125] Después de generar la lista de visualizaciones de objetivos de profundidad sin anclaje o en respuesta a determinar que los componentes de visualización de profundidad no deben incluirse en el sub-flujo de bits ("NO" de 310), el dispositivo CDN 16 puede marcar unidades VCL NAL aplicables y las unidades NAL de datos de relleno como "que se eliminarán del flujo de bits" (314). El dispositivo CDN 16 puede determinar las unidades NAL de datos de relleno y VAL aplicables de diversas maneras. Por ejemplo, el dispositivo CDN 16 puede determinar las unidades NAL de datos de relleno y VAL aplicables, realizando la operación de ejemplo de la FIG. 11 o la operación de ejemplo de la FIG. 12. Las FIGs. 11 y 12 se describen en detalle más adelante en esta divulgación.

[0126] A continuación, el dispositivo CDN 16 puede eliminar, del flujo de bits, cada unidad de acceso para la cual todas las unidades VCL NAL de la unidad de acceso están marcadas como "para ser eliminadas del flujo de bits" (316). Además, el dispositivo CDN 16 puede eliminar, del flujo de bits, todas las unidades VCL NAL y todas las unidades NAL de datos de relleno que están marcadas como "para ser eliminadas del flujo de bits" (318).

[0127] El dispositivo CDN 16 puede entonces determinar si la lista de índices de orden de visualización (VOIdxList) contiene solo un índice de orden de visualización (VOIdx) que es igual al índice de orden de visualización mínimo (minVOIdx) (320). Como se indicó anteriormente, la lista de índices de orden de visualización puede incluir los índices de orden de visualización de las visualizaciones marcadas como "requeridas para la textura del anclaje", "requeridas para la profundidad del anclaje", "requeridas para textura sin anclaje" y "requeridas para profundidad de anclaje". Cuando VOIdxList contiene solo un valor de VOIdx igual a minVOIdx, el sub-flujo de bits contiene solo la visualización base o solo un subconjunto temporal de la visualización base.

[0128] En respuesta a determinar que la lista de índices de orden de visualización (VOIdxList) contiene solo un índice de orden de visualización (VOIdx) que es igual al índice mínimo de orden de visualización (minVOIdx) ("SÍ" de 320), el dispositivo CDN 16 puede eliminar, del flujo de bits, todas las unidades NAL de prefijo y todas las unidades NAL de conjunto de parámetros de secuencia de subconjunto (es decir, unidades NAL con nal_unit_type igual a 14 o 15, respectivamente) (322).

[0129] Además, el dispositivo CDN 16 puede eliminar, del flujo de bits, cada unidad SEI NAL (es decir, unidades NAL con nal_unit_type igual a 6) en la que un primer mensaje SEI de la unidad SEI NAL tiene un tipo de carga útil en el rango de 36 a 44, inclusive (324). De esta forma, cuando VOIdxList contiene solo un valor de VOIdx que es igual a minVOIdx, el dispositivo CDN 16 puede eliminar las siguientes unidades NAL: todas las unidades NAL con nal_unit_type igual a 14 o 15 y todas las unidades NAL con nal_unit_type igual a 6 en las que el primer mensaje SEI tiene payloadType en el rango de 36 a 44, inclusive.

[0130] Los mensajes SEI que tienen un tipo de carga útil 36 incluyen información de descodificación paralela. Los mensajes SEI que tienen un tipo de carga útil 37 incluyen información de agrupamiento escalable MVC. Los mensajes SEI que tienen un tipo de carga útil 38 incluyen información de escalabilidad de visualización. Los mensajes SEI que tienen el tipo de carga útil 39 incluyen información de escena de múltiples visualizaciones. Los mensajes SEI que tienen un tipo de carga útil 40 incluyen información de adquisición de múltiples visualizaciones. Los mensajes SEI que tienen el tipo de carga útil 41 incluyen información de componente de visualización no requerida. Los mensajes SEI que tienen el tipo de carga útil 42 incluyen información de cambio de dependencia de visualización. Los mensajes SEI que tienen el tipo de carga útil 43 incluyen información de puntos de operación no presentes. Los mensajes SEI que tienen el tipo de carga útil 44 incluyen información de descodificador de referencia hipotética temporal (HRD) de visualización base.

[0131] Después de eliminar cada unidad SEI NAL en la que un primer mensaje SEI de la unidad SEI NAL tiene un tipo de carga útil en el rango de 36 a 44 o en respuesta a determinar que la lista de índices de orden de visualización (VOIdxList) no contiene solo un índice de orden de visualización (VOIdx) que es igual al índice de orden de visualización mínimo (minVOIdx) ("NO" de 320), el dispositivo CDN 16 puede determinar si los componentes de visualización de profundidad deben incluirse en el sub-flujo de bits (326). En respuesta a la determinación de que los

componentes de visualización de profundidad no deben incluirse en el sub-flujo de bits ("NO" de 326), el dispositivo CDN 16 puede eliminar, del flujo de bits, cada unidad SEI NAL (es decir, cada unidad NAL que tiene `nal_unit_type` igual a 6) en el que un primer mensaje SEI tiene un tipo de carga útil en el rango de 45 a 47, inclusive (328). Por ejemplo, cuando `depthPresentFlagTarget` es igual a 0, el dispositivo CDN 16 puede eliminar todas las unidades NAL con `nal_unit_type` igual a 6 en las que el primer mensaje SEI tiene un tipo de carga útil en el rango de 45 a 47, inclusive. Los mensajes SEI que tienen un tipo de carga útil 45 incluyen información de agrupamiento escalable 3DV. Los mensajes SEI que tienen el tipo de carga útil 46 incluyen información de escalabilidad de visualización 3D. Los mensajes SEI que tienen el tipo de carga útil 47 incluyen información de adquisición 3DV.

[0132] Después de eliminar cada unidad SEI NAL que tiene un primer mensaje SEI que tiene un tipo de carga útil en el rango de 45 a 47 (328) o en respuesta a determinar que componentes de visualización de profundidad deben ser incluidos en el sub-flujo de bits ("SÍ" de 326), el dispositivo CDN 16 puede llevar a cabo la continuación del proceso de extracción de sub-flujo de bits 298 mostrado en la FIG. 6.

[0133] La FIG. 6 es un diagrama de flujo que ilustra una continuación del proceso de extracción de sub-flujo de bits 298 de la FIG. 5. Como se muestra en el ejemplo de la FIG. 6, el dispositivo CDN 16 puede eliminar, del flujo de bits, todas las unidades SEI NAL (es decir, unidades NAL con `nal_unit_type` igual a 6) que contienen solo mensajes SEI que son parte de un mensaje de agrupamiento escalable MVC aplicable (330). Un mensaje de agrupamiento escalable de MVC puede ser un mensaje de agrupamiento escalable de MVC aplicable si un elemento sintáctico `operation_point_flag` del mensaje de agrupamiento escalable de MVC es igual a 0, un elemento sintáctico `all_view_components_in_au_flag` del mensaje de agrupamiento escalable de MVC es 0 y ninguno de los elementos sintácticos `sei_view_id[i]` del mensaje de agrupamiento escalable de MVC, para todos los *i* que están en el rango de 0 a `num_view_components_minus1`, inclusive, corresponden a un índice de orden de visualización (`VOIdx`) incluido en la lista de índices de orden de visualización (`VOIdxList`). Además, un mensaje de agrupamiento escalable de MVC puede ser un mensaje de agrupamiento escalable de MVC aplicable si un elemento sintáctico `operation_point_flag` del mensaje de agrupamiento escalable de MVC es igual a 1 y un elemento sintáctico `sei_op_temporal_id` del mensaje de agrupamiento escalable de MVC es mayor que `maxTId` o la lista de `sei_op_view_id[i]` especificada en el mensaje de agrupamiento escalable de MVC para todos los *i* en el rango de 0 a `num_view_components_op_minus1`, inclusive, no es un subconjunto de `viewIdTargetList`. `MaxTId` puede ser el `temporal_id` máximo de todas las unidades VCL NAL restantes. En otras palabras, un mensaje de agrupamiento escalable MVC puede ser un mensaje de agrupamiento escalable MVC aplicable si un `operation_point_flag` del mensaje de agrupamiento escalable MVC es igual a 1 y no es cierto que `sei_op_view_id[i]` para cualquier *i* en el rango de 0 a `num_view_components_minus1`, inclusive, es igual a un valor en `viewIdTargetList`.

[0134] De esta manera, el dispositivo CDN 16 puede dejar que `maxTId` sea el `temporal_id` máximo de todas las unidades VCL NAL restantes. Además, el dispositivo CDN 16 puede eliminar todas las unidades NAL con `nal_unit_type` igual a 6 que solo contienen mensajes SEI que son parte de un mensaje SEI de agrupamiento escalable MVC con cualquiera de las siguientes propiedades:

- `operation_point_flag` es igual a 0 y `all_view_components_in_au_flag` es igual a 0 y ninguno de `sei_view_id[i]` para todos los *i* en el rango de 0 a `num_view_components_minus1`, inclusive, corresponde a un valor `VOIdx` incluido en `VOIdxList`,
- `operation_point_flag` es igual a 1 y `sei_op_temporal_id` es mayor que `maxTId` o la lista de `sei_op_view_id[i]` para todos los *i* en el rango de 0 a `num_view_components_op_minus1`, inclusive, no es un subconjunto de `viewIdTargetList` (es decir, no es cierto que `sei_op_view_id[i]` para cualquier *i* en el rango de 0 a `num_view_components_op_minus1`, inclusive, es igual a un valor en `viewIdTargetList`).

[0135] A continuación, el dispositivo CDN 16 puede eliminar, de cada unidad NAL SEI en el flujo de bits, cada mensaje SEI de información de escalabilidad de visualización y cada mensaje SEI no presente de punto de operación, cuando está presente (332). Además, el dispositivo CDN 16 puede determinar si la lista de índices de orden de visualización (`VOIdxList`) contiene un índice de orden de visualización igual al índice mínimo de orden de visualización (`minVOIdx`) (334). En respuesta a determinar que la lista de índices de orden de visualización contiene un índice de orden de visualización igual al índice de orden de visualización mínimo ("SÍ" de 334), el dispositivo CDN 16 puede finalizar el proceso de extracción de sub-flujo de bits 298. Por otro lado, en respuesta a determinar que la lista de índices de orden de visualización no contiene un índice de orden de visualización igual al índice de orden de visualización mínimo ("NO" de 334), el dispositivo CDN 16 puede convertir la visualización con índice de orden de visualización igual al índice de orden de visualización mínimo en la visualización base del sub-flujo de bits extraído (336). Después de convertir la visualización, el dispositivo CDN 16 puede finalizar el proceso de extracción de sub-flujo de bits 298. Los datos restantes en el flujo de bits son un sub-flujo de bits que el dispositivo CDN 16 puede reenviar a otro dispositivo, tal como el dispositivo de destino 14. Por lo tanto, cuando `VOIdxList` no contiene un valor de `VOIdx` igual a `minVOIdx`, la visualización con `VOIdx` igual al valor mínimo de `VOIdx` incluido en `VOIdxList` se convierte en la visualización base del sub-flujo de bits extraído. En la sub-cláusula I.8.5.6 de 3DV compatible con MVC se describe un procedimiento informativo que describe los pasos clave de procesamiento para crear una visualización base.

[0136] Cuando la lista de índices visualización orden no contiene un índice de orden visualización igual al índice de orden de visualización mínimo, el sub-flujo de bits resultante generado por el proceso de extracción de sub-flujo de bits 298 no puede contener una visualización base que se ajusta a uno o más perfiles especificados en el anexo A del estándar H.264/AVC. Por lo tanto, cuando la lista de índices de orden de visualización no contiene un índice de orden de visualización igual al índice de orden de visualización mínimo, la visualización restante con el nuevo valor de índice de orden de visualización mínimo puede convertirse en la acción 336 para ser la nueva visualización base que se ajusta a uno o más perfiles especificados en el Anexo A y el Anexo H del estándar H.264/AVC.

[0137] La FIG. 7 es un diagrama de flujo que ilustra una operación de ejemplo 350 para determinar los identificadores de visualización de los componentes de visualización de textura de anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación. Como se describió anteriormente con respecto a la acción 302 de la FIG. 5, el dispositivo CDN 16 puede determinar los identificadores de visualización que están marcados como "requeridos para la textura de anclaje". El dispositivo CDN 16 puede determinar la lista de identificadores de visualización que están marcados como "requeridos para la textura de anclaje" invocando la operación 350 con un parámetro de entrada que especifica un identificador de visualización actual. En el contexto de la FIG. 7, esta divulgación puede referirse al índice de orden de visualización que corresponde al identificador de visualización actual como el "vOldx actual" o simplemente "vOldx". La operación 350 puede ser similar al proceso descrito en la sección H.8.5.1 del estándar H.264/AVC, sustituyendo el término "componente de visualización" por "componente de visualización de textura" y sustituyendo "requerido para el anclaje" por "requerido para la textura de anclaje".

[0138] Como se ilustra en el ejemplo de la FIG. 7, el dispositivo CDN 16 puede determinar que el identificador de visualización actual es requerido para descodificar un componente de visualización de textura de anclaje (352). Por ejemplo, el dispositivo CDN 16 puede marcar el identificador de visualización actual como "requerido para la textura de anclaje".

[0139] Además, el dispositivo CDN 16 puede determinar si tanto el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 (354). En respuesta a determinar que tanto el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 como el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual son iguales a 0 ("Sí" de 354), el dispositivo CDN 16 puede finalizar la operación 350.

[0140] Por otro lado, en respuesta a la determinación de que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es 0 o que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es igual a 0 ("NO" de 354), el dispositivo CDN 16 puede determinar si el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 (356).

[0141] En respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 ("Sí" de 356), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de textura de anclaje que son necesarios para la descodificación de cada componente de visualización de textura de anclaje en la lista 0 que está asociado con el índice de orden de visualización actual (358). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 350 para cada componente de visualización de textura de anclaje en la lista 0 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de textura de anclaje como entrada a la operación 350 cuando realiza de forma recursiva la operación 350 para el componente de visualización de textura de anclaje.

[0142] Por ejemplo, cuando $\text{num_anchor_refs_l0[vOldx]}$ no es igual a 0, el dispositivo CDN 16 puede invocar la operación 350 para cada componente de visualización de textura en $\text{anchor_ref_l0[vOldx][i]}$ para todos los i en el rango de 0 a $\text{num_anchor_l0[vOldx]} - 1$, inclusive, en orden ascendente de i . $\text{num_anchor_refs_l0[i]}$ y $\text{anchor_ref_l0[i][j]}$ son conjuntos de elementos sintácticos en una extensión de SPS MVC. $\text{num_anchor_refs_l0[i]}$ especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 (es decir, lista 0) en la descodificación de componentes de visualización de anclaje con índices de orden de visualización iguales a i . En otras palabras, $\text{num_anchor_refs_l0[i]}$ indica cuántos componentes de visualización inicialmente en la lista 0 se usan para la predicción entre visualizaciones de componentes de visualización de anclaje con índices de orden de visualización iguales a i . $\text{num_anchor_refs_l0[i]}$ especifica cuántos $\text{anchor_ref_l0[i][j]}$ especifican el identificador de visualización (view_id) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 (es decir, lista 0) en la descodificación de los componentes de visualización de anclaje con índice de orden de visualización igual a i .

[0143] De lo contrario, en respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 356) o después de realizar la acción 358, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 (360). En respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 360), el dispositivo CDN 16 puede finalizar la operación 350.

[0144] Sin embargo, en respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 360), el dispositivo CDN 16 puede determinar los identificadores de visualización de los componentes de visualización de textura de anclaje que son necesarios para la descodificación de cada componente de visualización de textura de anclaje en la lista 1 que está asociado con el índice de orden de visualización actual (362). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 350 para cada componente de visualización de textura de anclaje en la lista 1 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de textura de anclaje como entrada a la operación 350 cuando realiza de forma recursiva la operación 350 para el componente de visualización de textura de anclaje.

[0145] Por ejemplo, cuando `num_anchor_refs_l1[vOidx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 448 para cada componente de visualización de textura en `anchor_ref_l1[vOidx][i]` para todos los i en el rango de 0 a `num_anchor_refs_l1[vOidx] - 1`, inclusive, en orden ascendente de i . `num_anchor_refs_l1[i]` y `anchor_ref_l1[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC. `num_anchor_refs_l1[i]` especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización de anclaje con índices de orden de visualización iguales a i . `anchor_ref_l1[i][j]` especifica el identificador de visualización (`view_id`) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización de anclaje con índice de orden de visualización igual a i . Después de realizar la acción 362, el dispositivo CDN 16 puede finalizar la operación 350.

[0146] La FIG. 8 es un diagrama de flujo que ilustra una operación de ejemplo 400 para determinar los identificadores de visualización de los componentes de visualización de profundidad de anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación. Como se describió anteriormente con respecto a la acción 306 de la FIG. 5, el dispositivo CDN 16 puede determinar los identificadores de visualización que están marcados como "requeridos para profundidad de anclaje". El dispositivo CDN 16 puede determinar la lista de identificadores de visualización que están marcados como "requeridos para profundidad de anclaje" invocando la operación 400 con un parámetro de entrada que especifica un identificador de visualización actual. En el contexto de la FIG. 8, esta divulgación puede referirse al índice de orden de visualización que corresponde al identificador de visualización actual como el "vOidx actual" o simplemente "vOidx". La operación 400 puede ser similar al proceso descrito en la sección H.8.5.1 del estándar H.264/AVC, sustituyendo el término "componente de visualización" por "componente de visualización de profundidad" y sustituyendo "requerido para el anclaje" por "requerido para profundidad de anclaje".

[0147] Como se ilustra en el ejemplo de la FIG. 8, el dispositivo CDN 16 puede determinar que el identificador de visualización actual es requerido para descodificar un componente de visualización de profundidad de anclaje (402). Por ejemplo, el dispositivo CDN 16 puede marcar el identificador de visualización actual como "requerido para profundidad de anclaje".

[0148] Además, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 (404). En respuesta a la determinación de que tanto el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 ("SÍ" de 404), el dispositivo CDN 16 puede finalizar la operación 400.

[0149] Por otro lado, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es igual a 0 o que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es igual a 0 ("NO" de 404), el dispositivo CDN 16 puede determinar si el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 (406).

[0150] En respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor

que 0 ("SÍ" de 406), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de profundidad de anclaje que son necesarios para la descodificación de cada componente de visualización de profundidad de anclaje en la lista 0 que está asociado con el índice de orden de visualización actual (408). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 400 para cada componente de visualización de profundidad de anclaje en la lista 0 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de profundidad de anclaje como entrada a la operación 400 cuando realiza recursivamente la operación 400 para el componente de visualización de profundidad de anclaje.

[0151] Por ejemplo, cuando `num_anchor_refs_I0[vOldx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 400 para cada componente de visualización de profundidad en `anchor_ref_I0[vOldx][i]` para todos los i en el rango de 0 a `num_anchor_I0[vOldx] - 1`, inclusive, en orden ascendente de i . `num_anchor_refs_I0[i]` y `anchor_ref_I0[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC. `num_anchor_refs_I0[i]` especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList0` (es decir, lista 0) en los componentes de visualización de anclaje de descodificación con índices de orden de visualización iguales a i . `anchor_ref_I0[i][j]` especifica el identificador de visualización (`view_id`) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList0` (es decir, lista 0) en la descodificación de componentes de visualización de anclaje con índice de orden de visualización igual a i .

[0152] De lo contrario, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 406) o después de realizar la acción 408, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 (410). En respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 410), el dispositivo CDN 16 puede finalizar la operación 400.

[0153] Sin embargo, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones de anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 410), el dispositivo CDN 16 puede determinar los identificadores de visualización de los componentes de visualización de textura de anclaje que son necesarios para la descodificación de cada componente de visualización de profundidad de anclaje en la lista 1 que está asociado con el índice de orden de visualización actual (412). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 400 para cada componente de visualización de profundidad de anclaje en la lista 1 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de profundidad de anclaje como entrada a la operación 400 cuando realiza recursivamente la operación 400 para el componente de visualización de profundidad de anclaje.

[0154] Por ejemplo, cuando `num_anchor_refs_I1[vOldx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 400 para cada componente de visualización de profundidad en `anchor_ref_I1[vOldx][i]` para todos los i en el rango de 0 a `num_anchor_I1[vOldx] - 1`, inclusive, en orden ascendente de i . `num_anchor_refs_I1[i]` y `anchor_ref_I1[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC. `num_anchor_refs_I1[i]` especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización de anclaje con índices de orden de visualización iguales a i . `anchor_ref_I1[i][j]` especifica el identificador de visualización (`view_id`) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización de anclaje con índice de orden de visualización igual a i . Después de realizar la acción 412, el dispositivo CDN 16 puede finalizar la operación 400.

[0155] La FIG. 9 es un diagrama de flujo que ilustra una operación de ejemplo 450 para determinar identificadores de visualización de componentes de visualización de textura sin anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación. Como se describió anteriormente con respecto a la acción 308 de la FIG. 5, el dispositivo CDN 16 puede determinar los identificadores de visualización que están marcados como "requeridos para textura sin anclaje". El dispositivo CDN 16 puede determinar la lista de identificadores de visualización que están marcados como "requeridos para textura sin anclaje" invocando la operación 450 con un parámetro de entrada que especifica un identificador de visualización actual. En el contexto de la FIG. 9, esta divulgación puede referirse al índice de orden de visualización que corresponde al identificador de visualización actual como el "vOldx actual" o simplemente "vOldx". La operación 450 puede ser similar al proceso descrito en la sección H.8.5.2 del estándar H.264/AVC, sustituyendo el término "componente de visualización" por "componente de visualización de textura" y sustituyendo "requerido para sin anclaje" por "requerido para textura sin anclaje".

[0156] Como se ilustra en el ejemplo de la FIG. 9, el dispositivo CDN 16 puede determinar que el identificador de visualización actual es requerido para descodificar un componente de visualización de textura sin anclaje (452). Por

ejemplo, el dispositivo CDN 16 puede marcar el identificador de visualización actual como "requerido para textura sin anclaje".

5 **[0157]** Además, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de textura entre visualizaciones sin anclaje de la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 (454). En respuesta a la determinación de que tanto el número de componentes de visualización de textura entre visualizaciones sin anclaje de la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de
10 visualización de textura entre visualizaciones sin anclaje de la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 ("SÍ" de 454), el dispositivo CDN 16 puede finalizar la operación 450.

15 **[0158]** Por otro lado, en respuesta a la determinación de que el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es igual a 0 o que el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es igual a 0 ("NO" de 454), el dispositivo CDN 16 puede determinar si el número de componentes de visualización de textura entre visualizaciones sin anclaje la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 (456).

20 **[0159]** En respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones sin anclaje de la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 456), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de textura sin anclaje que son necesarios para la descodificación de cada componente de visualización de textura sin anclaje en la lista 0 que está asociado con el índice de orden de visualización actual (458). En algunos ejemplos, el
25 dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 450 para cada componente de visualización de textura sin anclaje en la lista 0 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de textura sin anclaje como entrada a la operación 450 cuando realiza recursivamente la operación 450 para el componente de visualización de textura sin anclaje.

30 **[0160]** Por ejemplo, cuando `num_non_anchor_refs_I0[vOIdx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 450 para cada componente de visualización de textura en `non_anchor_ref_I0[vOIdx][i]` para todos los i en el rango de 0 a `num_non_anchor_refs_I0[vOIdx] - 1`, inclusive, en orden ascendente de i . `num_non_anchor_refs_I0[i]` y `non_anchor_ref_I0[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC.
35 `num_non_anchor_refs_I0[i]` especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList0` (es decir, lista 0) en la descodificación de componentes de visualización sin anclaje con índices de orden de visualización iguales a i . `non_anchor_ref_I0[i][j]` especifica el identificador de visualización (`view_id`) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList0` (es decir, lista 0) en la descodificación de
40 componentes de visualización sin anclaje con índice de orden de visualización igual a i .

45 **[0161]** De lo contrario, en respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 456) o después de realizar una acción 458, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de textura entre visualizaciones sin anclaje de la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 (460). En respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 460), el dispositivo CDN 16 puede finalizar la
50 operación 450.

55 **[0162]** Sin embargo, en respuesta a determinar que el número de componentes de visualización de textura entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 460), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de textura sin anclaje que son necesarios para la descodificación de cada componente de visualización de textura sin anclaje en la lista 1 que está asociado con el índice de orden de visualización actual (462). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 450 para cada componente de visualización de textura sin anclaje en la lista 1 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de textura sin anclaje como entrada a la operación 450 cuando realiza recursivamente la operación 450 para el componente de visualización de textura sin anclaje.
60

65 **[0163]** Por ejemplo, cuando `num_non_anchor_refs_I1[vOIdx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 450 para cada componente de visualización de textura en `non_anchor_ref_I1[vOIdx][i]` para todos los i en el rango de 0 a `num_non_anchor_refs_I1[vOIdx] - 1`, inclusive, en orden ascendente de i . `num_non_anchor_refs_I1[i]` y `non_anchor_ref_I1[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC. `num_non_anchor_refs_I1[i]` especifica el número de componentes de visualización para la predicción entre

visualizaciones en la lista de imágenes de referencia inicial RefPicList1 (es decir, lista 1) en la descodificación de componentes de visualización sin anclaje con índices de orden de visualización iguales a i . $non_anchor_ref_l1[i][j]$ especifica el identificador de visualización (id de la visualización) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList1 (es decir, lista 1) en la descodificación de componentes de visualización sin anclaje con índice de orden de visualización igual a i . Después de realizar la acción 462, el dispositivo CDN 16 puede finalizar la operación 450.

[0164] La FIG. 10 es un diagrama de flujo que ilustra una operación de ejemplo 500 para determinar identificadores de visualización de componentes de visualización de profundidad sin anclaje requeridos, de acuerdo con una o más técnicas de esta divulgación. Como se describió anteriormente con respecto a la acción 312 de la FIG. 5, el dispositivo CDN 16 puede determinar los identificadores de visualización que están marcados como "requeridos para profundidad sin anclaje". El dispositivo CDN 16 puede determinar la lista de identificadores de visualización que están marcados como "requeridos para profundidad sin anclaje" invocando la operación 500 con un parámetro de entrada que especifica un identificador de visualización actual. En el contexto de la FIG. 10, esta divulgación puede referirse al índice de orden de visualización que corresponde al identificador de visualización actual como el "vOldx actual" o simplemente "vOldx". La operación 500 puede ser similar al proceso descrito en la sección H.8.5.2 del estándar H.264/AVC, sustituyendo el término "componente de visualización" por "componente de visualización de profundidad" y sustituyendo "requerido para sin anclaje" por "requerido para profundidad sin anclaje".

[0165] Como se ilustra en el ejemplo de la FIG. 10, el dispositivo CDN 16 puede determinar que el identificador de visualización actual es requerido para descodificar un componente de visualización de profundidad sin anclaje (502). Por ejemplo, el dispositivo CDN 16 puede marcar el identificador de visualización actual como "requerido para profundidad sin anclaje".

[0166] Además, el dispositivo CDN 16 puede determinar si tanto el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 (504). En respuesta a determinar que tanto el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es igual a 0 y el número de componentes de visualización de profundidad entre visualizaciones sin anclaje de la lista 1 que están asociados con el índice de orden de visualización actual es igual a 0 ("SÍ" de 504), el dispositivo CDN 16 puede finalizar la operación 500.

[0167] Por otro lado, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje de la lista 0 que están asociados con el índice de orden de visualización actual no es igual a 0 o que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es igual a 0 ("NO" de 504), el dispositivo CDN 16 puede determinar si el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 (506).

[0168] En respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 506), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de profundidad sin anclaje que son necesarios para la descodificación de cada componente de visualización de profundidad sin anclaje en la lista 0 que está asociado con el índice de orden de visualización actual (508). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 500 para cada componente de visualización de profundidad sin anclaje en la lista 0 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de profundidad sin anclaje como entrada a la operación 500 cuando realiza recursivamente la operación 500 para el componente de visualización de profundidad sin anclaje.

[0169] Por ejemplo, cuando $num_non_anchor_refs_l0[vOldx]$ no es igual a 0, el dispositivo CDN 16 puede invocar la operación 500 para cada componente de visualización de profundidad en $non_anchor_ref_l0[vOldx][i]$ para todos los i en el rango de 0 a $num_non_anchor_l0[vOldx] - 1$, inclusive, en orden ascendente de i . $num_non_anchor_refs_l0[i]$ y $non_anchor_ref_l0[i][j]$ son conjuntos de elementos sintácticos en una extensión de SPS MVC. $num_non_anchor_refs_l0[i]$ especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 (es decir, lista 0) en la descodificación de componentes de visualización sin anclaje con índices de orden de visualización iguales a i . $non_anchor_ref_l0[i][j]$ especifica el identificador de visualización (view_id) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial RefPicList0 (es decir, lista 0) en la descodificación de componentes de visualización sin anclaje con índice de orden de visualización igual a i .

[0170] De lo contrario, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 0 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 506) o después de realizar la acción 508, el dispositivo CDN 16 puede determinar si el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 1 que están asociados

con el índice de orden de visualización actual es mayor que 0 (510). En respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje en la lista 1 que están asociados con el índice de orden de visualización actual no es mayor que 0 ("NO" de 510), el dispositivo CDN 16 puede finalizar la operación 500.

[0171] Sin embargo, en respuesta a determinar que el número de componentes de visualización de profundidad entre visualizaciones sin anclaje de la lista 1 que están asociados con el índice de orden de visualización actual es mayor que 0 ("SÍ" de 510), el dispositivo CDN 16 puede determinar los identificadores de visualización de componentes de visualización de textura sin anclaje que son necesarios para la descodificación de cada componente de visualización de profundidad sin anclaje en la lista 1 que está asociado con el índice de orden de visualización actual (512). En algunos ejemplos, el dispositivo CDN 16 puede hacerlo ejecutando recursivamente la operación 500 para cada componente de visualización de profundidad sin anclaje en la lista 1 que está asociado con el índice de orden de visualización actual. En dichos ejemplos, el dispositivo CDN 16 puede proporcionar el identificador de visualización de un componente de visualización de profundidad sin anclaje como entrada a la operación 500 cuando realiza recursivamente la operación 500 para el componente de visualización de profundidad sin anclaje.

[0172] Por ejemplo, cuando `num_non_anchor_refs_l1[vOldx]` no es igual a 0, el dispositivo CDN 16 puede invocar la operación 500 para cada componente de visualización de profundidad en `non_anchor_ref_l1[vOldx][i]` para todos los i en el rango de 0 a `num_non_anchor_refs_l1[vOldx] - 1`, inclusive, en orden ascendente de i . `num_non_anchor_refs_l1[i]` y `non_anchor_ref_l1[i][j]` son conjuntos de elementos sintácticos en una extensión de SPS MVC. `num_non_anchor_refs_l1[i]` especifica el número de componentes de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización sin anclaje con índices de orden de visualización iguales a i . `non_anchor_ref_l1[i][j]` especifica el identificador de visualización (id de la visualización) del j -ésimo componente de visualización para la predicción entre visualizaciones en la lista de imágenes de referencia inicial `RefPicList1` (es decir, lista 1) en la descodificación de componentes de visualización sin anclaje con índice de orden de visualización igual a i . Después de realizar la acción 512, el dispositivo CDN 16 puede finalizar la operación 500.

[0173] La FIG. 11 es un diagrama de flujo que ilustra una primera operación de ejemplo 550 para marcar las unidades VCL NAL y las unidades NAL de datos de relleno de manera que se eliminen de un flujo de bits, de acuerdo con una o más técnicas de esta divulgación. El dispositivo CDN 16 puede realizar la operación de la FIG. 11 como al menos parte de la realización de la acción 316 en el ejemplo de la FIG. 5. La siguiente descripción de la FIG. 11 puede ser aplicable a unidades VCL NAL y unidades NAL de datos de relleno.

[0174] Como se ilustra en el ejemplo de la FIG. 11, el dispositivo CDN 16 puede determinar si un identificador de prioridad (`priority_id`) de la unidad NAL es mayor que `pldTarget` (552). Como se analizó anteriormente, la extensión de cabecera de la unidad MVC NAL de la unidad NAL puede incluir el elemento sintáctico de id de prioridad. `pldTarget` es un parámetro que se proporciona al proceso de extracción de sub-flujo de bits, tal como el proceso de extracción de sub-flujo de bits 298 de la FIG. 5. `pldTarget` puede identificar un nivel de prioridad objetivo. En respuesta a determinar que el identificador de prioridad de la unidad NAL es mayor que `pldTarget` ("SÍ" de 552), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y las unidades NAL de datos de relleno como "para eliminar del flujo de bits" donde los elementos sintácticos `priority_id` de las unidades NAL son mayores que `pldTarget`.

[0175] Por otra parte, en respuesta a determinar que el identificador de prioridad de la unidad NAL no es mayor que `pldTarget` ("NO" de 552), el dispositivo CDN 16 puede determinar si un identificador temporal (`temporal_id`) de la unidad NAL es mayor que `tldTarget` (556). Como se analizó anteriormente, la extensión de cabecera de la unidad MVC NAL de la unidad NAL puede incluir el elemento sintáctico id temporal y `tldTarget` es un parámetro que se proporciona al proceso de extracción de sub-flujo de bits, tal como el proceso de extracción de sub-flujo de bits 298. `tldTarget` puede identificar un nivel temporal objetivo. En respuesta a la determinación de que el identificador temporal de la unidad NAL es mayor que `tldTarget` ("SÍ" de 556), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y las unidades NAL de datos de relleno como "para eliminar del flujo de bits" donde los elementos sintácticos `temporal_id` de las unidades NAL son mayores que `tldTarget`.

[0176] En respuesta a la determinación de que el identificador temporal de la unidad NAL no es mayor que `tldTarget` ("NO" de 556), el dispositivo CDN 16 puede, de acuerdo con las técnicas de esta divulgación, determinar si la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto a un tipo de unidad NAL de componente de visualización de profundidad, y un identificador de visualización (`view_id`) de la unidad NAL no está marcado como "requerido para textura de anclaje" (558). El dispositivo CDN 16 puede determinar, basándose en un indicador de imagen de anclaje (`anchor_pic_flag`) de la unidad NAL, si la unidad NAL pertenece a una imagen de anclaje. Las unidades NAL que pertenecen al tipo de unidad NAL del componente de visualización de profundidad pueden incluir extensiones de fragmentos codificados para los componentes de visualización de profundidad. En algunos ejemplos, las unidades NAL que tienen `nal_unit_type` igual a 21 pertenecen al tipo de unidad NAL de componente de visualización de profundidad.

[0177] En respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para textura de anclaje" ("SÍ" de 558), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si el `anchor_pic_flag` de la unidad NAL es igual a 1, el `nal_unit_type` de la unidad NAL no es 21, y el `view_id` de la unidad NAL no está marcado como "requerido para textura de anclaje".

[0178] Los identificadores de visualización marcados como "requeridos para textura de anclaje" están en la lista de visualizaciones de objetivos de textura de anclaje. Así, el dispositivo CDN 16 puede determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece a un tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura de anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV, en la que las unidades NAL que pertenecen al tipo de unidad NAL de componente de visualización de profundidad encapsulan extensiones de fragmentos codificados para componentes de visualización de profundidad. Por ejemplo, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL no es igual a 21, y un elemento sintáctico identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de textura de anclaje.

[0179] Sin embargo, en respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para textura de anclaje" ("NO" de 558), el dispositivo CDN 16 puede, de acuerdo con las técnicas de esta divulgación, determinar si la unidad NAL pertenece a una imagen sin anclaje, el tipo de unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente visualización de profundidad y un identificador de visualización (`id` de la visualización) de la unidad NAL no están marcados como "requeridos para una textura sin anclaje" (560).

[0180] En respuesta a determinar que la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para textura sin anclaje" ("SÍ" de 560), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 0, `nal_unit_type` de la unidad NAL no es 21, y el `view_id` de la unidad NAL no está marcado como "requerido para textura sin anclaje".

[0181] Los identificadores de visualización marcados como "requeridos para textura sin anclaje" están en la lista de objetivos de la visualización de textura sin anclaje. Así, el dispositivo CDN 16 puede determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura sin anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV. Por ejemplo, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL no es igual a 21, y el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de textura sin anclaje.

[0182] En respuesta a determinar que la unidad NAL no pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para textura sin anclaje" ("NO" de 560), el dispositivo CDN 16 puede determinar si la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y un identificador de visualización (`view_id`) de la unidad NAL no está marcado como "requerido para profundidad del anclaje" (562).

[0183] En respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para profundidad de anclaje" ("SÍ" de 562), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 1, `nal_unit_type` de la unidad NAL es 21, y la `id` de la visualización de la unidad NAL no está marcada como "requerida para la profundidad del anclaje".

[0184] Los identificadores de visualización marcados como "requeridos para profundidad de anclaje" se encuentran en la lista de visualizaciones de objetivos de profundidad de anclaje. Así, el dispositivo CDN 16 puede determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece a

un tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de profundidad de anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV, en la que las unidades NAL que pertenecen al tipo de unidad NAL de componente de visualización de profundidad encapsulan extensiones de fragmentos codificados para componentes de visualización de profundidad.

5 Por ejemplo, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL es igual a 21, y un elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad de anclaje.

10 **[0185]** Además, en respuesta a determinar que la unidad NAL no pertenece a una imagen de anclaje, la unidad NAL no pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para profundidad de anclaje" ("NO" de 562), el dispositivo CDN 16 puede determinar si la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad y un identificador de visualización (identificación de visualización) de la unidad NAL no está marcada como "requerida para la profundidad sin anclaje" (564).

15 **[0186]** En respuesta a determinar que la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para profundidad sin anclaje" ("SÍ" de 564), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 0, `nal_unit_type` de la unidad NAL es 21, y `view_id` de la unidad NAL no está marcado como "requerido para profundidad sin anclaje".

20 **[0187]** Los identificadores de visualización marcados como "requeridos para profundidad sin anclaje" son identificadores de visualización en la lista de visualizaciones de objetivos de profundidad sin anclaje. Así, el dispositivo CDN 16 puede determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de profundidad sin anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV. Por ejemplo, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL es igual a 21, y el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad sin anclaje.

25 **[0188]** En respuesta a determinar que la unidad NAL no pertenece a una imagen sin anclaje, la unidad NAL no pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para profundidad sin anclaje" ("NO" de 564), el dispositivo CDN 16 puede determinar si un indicador de referencia NAL (`nal_ref_idc`) de la unidad NAL es igual a 0, un indicador entre visualizaciones (indicador entre visualizaciones) de la unidad NAL es igual a 0, y un identificador de visualización de la unidad NAL es igual a cualquier valor en la lista de objetivos de identificador de visualización (`viewIdTargetList`) (566).

30 **[0189]** En respuesta a la determinación de que el indicador de referencia NAL de la unidad NAL es igual a 0, un indicador entre visualizaciones de la unidad NAL es igual a 0, y un identificador de visualización de la unidad NAL es igual a cualquier valor en la lista de objetivos de identificador de visualización ("SÍ" de 566), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para ser eliminada del flujo de bits" si el `nal_ref_idc` de la unidad NAL es igual a 0, el indicador `inter_view` es igual a 0 y la `id` de visualización de la unidad NAL no es igual a ningún valor en la lista `viewIdTargetList`.

35 **[0190]** Así, el dispositivo CDN 16 puede determinar que una unidad NAL debe eliminarse de un flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo unidad NAL en la cabecera de la unidad NAL es igual a 21, y un elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad de anclaje. Además, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico tipo unidad NAL en la cabecera del NAL unidad es igual a 21, y el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad sin anclaje.

40 **[0191]** Por otro lado, en respuesta a la determinación de que el indicador de referencia NAL de la unidad NAL no es igual a 0, el indicador entre visualizaciones de la unidad NAL no es igual a 0, o el identificador de visualización de la unidad NAL no es igual a ningún valor en la lista de objetivos de identificador de visualización ("NO" de 566), el dispositivo CDN 16 puede determinar si el tipo de unidad NAL de la unidad NAL es igual a 21 y un objetivo de indicador

presente de profundidad (depthPresentFlagTarget) es igual a 0 (568). En respuesta a la determinación de que el tipo de unidad NAL de la unidad NAL es igual a 21 y el objetivo de indicador de profundidad presente es igual a 0 ("SÍ" de 568), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (554). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para ser eliminada del flujo de bits" si nal_unit_type de la unidad NAL es igual a 21 y depthPresentFlagTarget es igual a 0.

[0192] En respuesta a determinar que el tipo de unidad NAL de la unidad NAL no es igual a 21 o el objetivo de indicador de profundidad presente no es igual a 0 ("NO" de 568), el dispositivo CDN 16 puede determinar si el objetivo de indicador presente de profundidad es igual a 1 (570). En respuesta a la determinación de que el objetivo del indicador presente de profundidad es igual a 1 ("SÍ" de 570), el dispositivo CDN 16 puede marcar a la unidad NAL como "para ser eliminada del flujo de bits" (554). De lo contrario, en respuesta a determinar que el indicador presente de profundidad no es igual a 1 ("NO" de 570), el dispositivo CDN 16 puede no marcar la unidad NAL como "para ser eliminada del flujo de bits" (572). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para ser eliminada del flujo de bits" si depthPresentFlagTarget es igual a 1.

[0193] La FIG. 12 es un diagrama de flujo que ilustra una segunda operación de ejemplo 600 para marcar las unidades VCL NAL y las unidades NAL de datos de relleno como para ser eliminadas de un flujo de bits, de acuerdo con una o más técnicas de esta divulgación. El dispositivo CDN 16 puede realizar la operación de ejemplo de la FIG. 12 en lugar de realizar la operación de ejemplo de la FIG. 11. La operación 600 puede tomar en consideración el elemento sintáctico del indicador sintáctico de uso (use_texture_flag) . Es decir, cuando el dispositivo CDN 16 realiza una operación de extracción de sub-flujo de bits que usa la operación 600 para determinar unidades VCL NAL y unidades NAL de datos de relleno para eliminar del flujo de bits, el dispositivo CDN puede determinar, basándose al menos en parte en un elemento sintáctico de indicador de textura de uso en una cabecera de una unidad NAL, si eliminar la unidad NAL de un flujo de bits 3DV. Como se indicó previamente, el elemento sintáctico de indicador de textura de uso puede indicar si un componente de visualización de profundidad encapsulado por la unidad NAL es descodificable sin descodificar un componente de visualización de textura que corresponde al componente de visualización de profundidad. El dispositivo CDN 16 puede realizar la operación de la FIG. 12 como al menos parte de la realización de la acción 316 en el ejemplo de la FIG. 5. La siguiente descripción de la FIG. 12 puede ser aplicable a unidades VCL NAL y unidades NAL de datos de relleno.

[0194] Como se ilustra en el ejemplo de la FIG. 12, el dispositivo CDN 16 puede determinar si un identificador de prioridad (priority_id) de la unidad NAL es mayor que pldTarget (602). Como se analizó anteriormente, la extensión de cabecera de la unidad MVC NAL de la unidad NAL incluye el elemento sintáctico de id de prioridad. pldTarget es un parámetro que se proporciona al proceso de extracción de sub-flujo de bits, tal como el proceso de extracción de sub-flujo de bits 298 de la FIG. 5. pldTarget puede identificar un nivel de prioridad objetivo. En respuesta a determinar que el identificador de prioridad de la unidad NAL es mayor que pldTarget ("SÍ" de 602), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para ser eliminada del flujo de bits" si la id de prioridad de la unidad NAL es mayor que pldTarget.

[0195] Por otra parte, en respuesta a determinar que el identificador de prioridad de la unidad NAL no es mayor que pldTarget ("NO" de 602), el dispositivo CDN 16 puede determinar si un identificador temporal (temporal_id) de la unidad NAL es mayor que tldTarget (606). Como se analizó anteriormente, la extensión de cabecera de la unidad MVC NAL de la unidad NAL incluye el elemento sintáctico temporal_id y la tldTarget es un parámetro que se proporciona al proceso de extracción de sub-flujo de bits, como el proceso de extracción de sub-flujo de bits 298. tldTarget puede identificar un nivel temporal de objetivo. En respuesta a la determinación de que el identificador temporal de la unidad NAL es mayor que tldTarget ("SÍ" de 606), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para ser eliminada del flujo de bits" si el temporal_id de la unidad NAL es mayor que tldTarget.

[0196] En respuesta a determinar que el identificador temporal de la unidad NAL no es mayor que tldTarget ("NO" de 606), el dispositivo CDN 16 puede, de acuerdo con las técnicas de esta divulgación, determinar si la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, un identificador de visualización (id_view) de la unidad NAL no está marcado como "requerido para textura de anclaje" y el indicador de textura de uso (use_texture_flag) es 0 o el identificador de visualización no está marcado como "requerido para profundidad de anclaje" (608). El dispositivo CDN 16 puede determinar, basándose en un indicador de imagen de anclaje (anchor_pic_flag) de la unidad NAL, si la unidad NAL pertenece a una imagen de anclaje. Las unidades NAL que pertenecen al tipo de unidad NAL del componente de visualización de profundidad pueden incluir extensiones de fragmentos codificados para los componentes de visualización de profundidad. En algunos ejemplos, las unidades NAL que tienen nal_unit_type igual a 21 pertenecen al tipo de unidad NAL de componente de visualización de profundidad.

[0197] En respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, el identificador de visualización de la unidad NAL no está marcado como "requerido para textura de anclaje", y el indicador de textura de uso (use_texture_flag) es 0 o el identificador de visualización no está marcado como "requerido para profundidad

de anclaje" ("Sí" de 608), el dispositivo CDN 16 puede marcar la unidad NAL como "para eliminar del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 1, `nal_unit_type` de la unidad NAL no es 21, y las dos condiciones siguientes se cumplen: el `view_id` de la unidad NAL no está marcado como "requerido para textura del anclaje" y `use_texture_flag` es 0 o la id de la visualización de la unidad NAL no está marcada como "requerido para la profundidad del anclaje".

[0198] Sin embargo, en respuesta a determinar que la unidad NAL no pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, el identificador de visualización de la unidad NAL está marcado como "requerido para textura de anclaje", o el indicador de textura de uso no es 0 y el identificador de visualización está marcado como "requerido para profundidad de anclaje" ("NO" de 608), el dispositivo CDN 16 puede, de acuerdo con las técnicas de esta divulgación, determinar si la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, un identificador de visualización (id de visualización) de la unidad NAL no está marcado como "requerido para textura sin anclaje" y el indicador de textura de uso es 0 o el identificador de visualización no está marcado como "requerido para profundidad sin anclaje" (610).

[0199] En respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece a un tipo de unidad NAL distinto al tipo de unidad NAL de componente de visualización de profundidad, el identificador de visualización de la unidad NAL no está marcado como "requerido para textura sin anclaje", y el indicador de textura de uso es 0 o el identificador de visualización no está marcado como "requerido para profundidad sin anclaje" ("Sí" de 610), el dispositivo CDN 16 puede marcar la unidad NAL como "para eliminar del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 0, `nal_unit_type` de la unidad NAL no es 21, y se cumplen las dos condiciones siguientes: `view_id` de la unidad NAL no está marcado como "requerido para textura sin anclaje" y `use_texture_flag` es 0 o la id de la visualización de la unidad NAL no está marcada como "requerida para profundidad sin anclaje".

[0200] En respuesta a determinar que la unidad NAL no pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, el identificador de visualización de la unidad NAL está marcado como "requerido para textura sin anclaje", o el indicador de textura de uso no es 1 y el identificador de visualización está marcado como "requerido para profundidad sin anclaje" ("NO" de 610), el dispositivo CDN 16 puede determinar si la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y un identificador de visualización (id de la visualización) de la unidad NAL no está marcado como "requerido para profundidad de anclaje" (612).

[0201] En respuesta a determinar que la unidad NAL pertenece a una imagen de anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para profundidad de anclaje" ("Sí" de 612), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar una unidad VCL NAL o una unidad NAL de datos de relleno como "para eliminar del flujo de bits" si `anchor_pic_flag` de la unidad NAL es igual a 1, `nal_unit_type` de la unidad NAL es 21, y el `view_id` de la unidad NAL no está marcado como "requerido para profundidad del anclaje".

[0202] Así, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo unidad NAL en la cabecera de la unidad NAL es igual a 21, un elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL no es igual a un identificador de visualización de cualquier visualización en la lista de objetivos de textura de anclaje, y el elemento sintáctico de indicador de textura de uso en la cabecera de la unidad NAL es igual a 0 o el elemento sintáctico de visualización en la cabecera de la unidad NAL no es igual a un identificador de visualización de cualquier visualización en la lista de objetivos de profundidad de anclaje.

[0203] Además, en respuesta a determinar que la unidad NAL no pertenece a una imagen de anclaje, la unidad NAL no pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para profundidad de anclaje" ("NO" de 612), el dispositivo CDN 16 puede determinar si la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad y un identificador de visualización (identificación de visualización) de la unidad NAL no está marcada como "requerida para la profundidad sin anclaje" (614).

[0204] En respuesta a determinar que la unidad NAL pertenece a una imagen sin anclaje, la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y el identificador de visualización de la unidad NAL no está marcado como "requerido para profundidad sin anclaje" ("Sí" de 614), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y unidades NAL de datos de relleno para las cuales el indicador de imagen de anclaje de la unidad NAL es igual a 0, `nal_unit_type` de la unidad NAL es igual a 21 y el identificador de visualización de la

unidad NAL no está marcado como "requerido para la profundidad sin anclaje" como "para ser eliminadas del flujo de bits".

5 **[0205]** Así, el dispositivo CDN 16 puede determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL es igual a 21, el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL no es igual a un identificador de visualización de cualquier visualización en la lista de objetivos de textura de anclaje, y el elemento sintáctico de indicador de textura de uso en la cabecera de la unidad NAL es igual a 0 o el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL no es igual a un identificador de visualización de cualquier visualización en la lista de objetivos de profundidad sin anclaje.

15 **[0206]** En respuesta a determinar que la unidad NAL no está en una imagen sin anclaje, la unidad NAL no pertenece al tipo de unidad NAL de componente de visualización de profundidad, o el identificador de visualización de la unidad NAL está marcado como "requerido para profundidad sin anclaje" ("NO" de 614), el dispositivo CDN 16 puede determinar si un indicador de referencia NAL (nal_ref_idc) de la unidad NAL es igual a 0, un indicador entre visualizaciones (indicador entre visualizaciones) de la unidad NAL es igual a 0, y un identificador de visualización de la unidad NAL es igual a cualquier valor en la lista de objetivos de identificador de visualización (viewIdTargetList) (616). En respuesta a determinar que el indicador de referencia NAL de la unidad NAL es igual a 0, un indicador entre visualizaciones de la unidad NAL es igual a 0, y un identificador de visualización de la unidad NAL es igual a cualquier valor en la lista de objetivos de identificador de visualización ("SÍ" de 616), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y unidades NAL de datos de relleno para las cuales el indicador de referencia NAL de la unidad NAL es igual a 0, un indicador entre visualizaciones de la unidad NAL es igual a 0, y un identificador de visualización de la unidad NAL es igual a cualquier valor en la lista de objetivos de identificador de visualización como "para ser eliminadas del flujo de bits".

30 **[0207]** Por otro lado, en respuesta a la determinación de que el indicador de referencia NAL de la unidad NAL no es igual a 0, el indicador entre visualizaciones de la unidad NAL no es igual a 0, o el identificador de visualización de la unidad NAL no es igual a ningún valor en la lista de objetivos de identificador de visualización ("NO" de 616), el dispositivo CDN 16 puede determinar si el tipo de unidad NAL de la unidad NAL es igual a 21 y un objetivo de indicador presente de profundidad (depthPresentFlagTarget) es igual a 0 (618). En respuesta a la determinación de que el tipo de unidad NAL de la unidad NAL es igual a 21 y el objetivo de indicador de profundidad presente es igual a 0 ("SÍ" de 618), el dispositivo CDN 16 puede marcar la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y unidades NAL de datos de relleno para las cuales el tipo de unidad NAL de la unidad NAL es igual a 21 y el objetivo de indicador presente de profundidad es igual a 0 "para ser eliminadas del flujo de bits".

40 **[0208]** En respuesta a determinar que el tipo de unidad NAL de la unidad NAL no es igual a 21 o el objetivo de indicador de profundidad presente no es igual a 0 ("NO" de 618), el dispositivo CDN 16 puede determinar si el objetivo de indicador presente de profundidad es igual a 1 (620). En respuesta a la determinación de que el objetivo del indicador presente de profundidad es igual a 1 ("SÍ" de 620), el dispositivo CDN 16 puede marcar a la unidad NAL como "para ser eliminada del flujo de bits" (604). De esta forma, el dispositivo CDN 16 puede marcar todas las unidades VCL NAL y las unidades NAL de datos de relleno para las cuales depthPresentFlagTarget es igual a 1 como "para ser eliminadas del flujo de bits". De lo contrario, en respuesta a determinar que el indicador presente de profundidad no es igual a 1 ("NO" de 620), el dispositivo CDN 16 puede no marcar la unidad NAL como "para ser eliminada del flujo de bits" (622).

50 **[0209]** En otros ejemplos, las cabeceras de unidad NAL no incluyen elementos sintácticos use_texture_flag. En dichos ejemplos, el dispositivo CDN 16 puede usar elementos sintácticos depth_to_view_flag de componentes de visualización de textura para obtener el valor de use_texture_flag. En otras palabras, use_texture_flag no se señala en la cabecera de la unidad NAL y depth_to_view_flag en el componente de visualización de textura se usa para obtener el valor de use_texture_flag. En algunos ejemplos, use_texture_flag se obtiene para ser igual a depth_to_view_flag.

55 **[0210]** La FIG. 13 es un diagrama conceptual que ilustra un ejemplo de orden de decodificación MVC. En el ejemplo de la FIG. 13, cada cuadrado corresponde a un componente de visualización. Cada uno de los componentes de visualización puede incluir un componente de visualización de textura y un componente de visualización de profundidad. Las columnas de cuadrados corresponden a las unidades de acceso. Cada unidad de acceso puede definirse para contener las imágenes codificadas de todas las visualizaciones de un instante de tiempo. Las filas de cuadrados corresponden a las visualizaciones. En el ejemplo de la FIG. 13, las unidades de acceso están etiquetadas T0 ... T7 y las visualizaciones están etiquetadas S0 ... S7. Debido a que cada componente de visualización de una unidad de acceso se decodifica antes de cualquier componente de visualización de la siguiente unidad de acceso, el orden de decodificación de la FIG. 13 puede denominarse codificación primera en el tiempo. Como se muestra en el ejemplo de la FIG. 13, el orden de decodificación de las unidades de acceso puede no ser idéntico al orden de salida o visualización de las visualizaciones.

[0211] Como se menciona anteriormente, un índice de orden de visualización es un índice que indica el orden de descodificación de los componentes de visualización en una unidad de acceso. Por lo tanto, en el ejemplo de la FIG. 13, el índice de orden de visualización de los componentes de visualización en la visualización S0 puede ser 0, el índice de orden de visualización de los componentes de visualización en la visualización S1 puede ser 1, el índice de orden de visualización de los componentes de visualización en la visualización S2 puede ser 2, etc.

[0212] La FIG. 14 es un diagrama conceptual que ilustra un ejemplo de estructura de predicción entre visualizaciones y temporal de MVC. Una estructura de predicción MVC típica (que incluye tanto la predicción entre imágenes dentro de cada visualización como la predicción entre visualizaciones) para la codificación de vídeo de múltiples visualizaciones se muestra en la FIG. 14. En el ejemplo de la FIG. 14, cada cuadrado corresponde a un componente de visualización. Cada uno de los componentes de visualización puede incluir un componente de visualización de textura y un componente de visualización de profundidad. Los cuadrados etiquetados como "I" son componentes de visualización sometidas a predicción intra. Los cuadrados etiquetados como "P" son componentes de visualización con predicción inter unidireccional. Los cuadrados etiquetados "B" y "b" son componentes de visualización con predicción inter bidireccional. Los cuadrados etiquetados como "b" pueden usar cuadrados etiquetados como "B" como imágenes de referencia. Las predicciones se indican con flechas, con el objeto que apunta que usa el objeto apuntado para referencia de predicción. Por ejemplo, una flecha que apunta desde un primer cuadrado a un segundo cuadrado indica que el primer cuadrado está disponible en la predicción inter como imagen de referencia para el segundo cuadrado. Como se indica por las flechas verticales en la FIG. 14, los componentes de visualización en diferentes visualizaciones de la misma unidad de acceso pueden estar disponibles como imágenes de referencia. El uso de un componente de visualización de una unidad de acceso como imagen de referencia para otra componente de visualización de la misma unidad de acceso puede denominarse predicción entre visualizaciones.

[0213] Un codificador de vídeo puede realizar un proceso de construcción de lista de imágenes de referencia para disponer de manera flexible referencias de predicción temporales y de visualización. La realización del proceso de construcción de la lista de imágenes de referencia puede proporcionar no solo posibles ganancias de eficiencia de codificación sino también capacidad de recuperación de errores, ya que las secciones de imagen de referencia y los mecanismos de imagen redundantes se pueden extender a la dimensión de visualización.

[0214] La construcción de la lista de imágenes de referencia puede incluir los siguientes pasos. En primer lugar, el codificador de vídeo puede aplicar el proceso de inicialización de la lista de imágenes de referencia para imágenes de referencia temporales (dentro de las visualizaciones) como se especifica en un estándar H.264/AVC, sin el uso de imágenes de referencia de otras visualizaciones. En segundo lugar, el codificador de vídeo puede agregar las imágenes de referencia entre visualizaciones al final de la lista en el orden en que aparecen las imágenes de referencia entre visualizaciones en la extensión SPS MVC. En tercer lugar, el codificador de vídeo aplica el proceso de modificación de la lista de imágenes de referencia (RPLM) para imágenes de referencia tanto centro de las visualizaciones como entre visualizaciones. El codificador de vídeo puede identificar imágenes de referencia entre visualizaciones en comandos RPLM por sus valores de índice como se especifica en la extensión MVC SPS.

[0215] 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, 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.

[0216] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda utilizarse para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe debidamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde 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 los anteriores también deben incluirse dentro del alcance de los medios legibles por ordenador.

- 5 **[0217]** 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 in situ (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, el término «procesador», como se utiliza en el presente documento, puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Asimismo, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.
- 10
- 15 **[0218]** Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, que incluyen un teléfono inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por medio de un grupo de unidades de hardware interoperativas, que incluyen uno o más procesadores como los descritos anteriormente, conjuntamente con software y/o firmware adecuados.
- 20
- 25 **[0219]** Se han descrito varios ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

- 5 1. Un procedimiento para extraer un sub-flujo de bits de un vídeo tridimensional, 3DV, flujo de bits que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados, comprendiendo el procedimiento:

10 determinar una primera lista que indica visualizaciones en el flujo de bits 3DV, con las visualizaciones indicadas por la primera lista que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos, con la pluralidad de visualizaciones de objetivos que comprende un subconjunto de las visualizaciones en el flujo de bits 3DV que se puede descodificar a partir del sub-flujo de bits; determinar una segunda lista independiente que indica visualizaciones en el flujo de bits 3DV, con las visualizaciones indicadas por la segunda lista que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos, con la primera lista y la segunda lista que indican subconjuntos diferentes de la visualizaciones en el flujo de bits 3DV; y

15 determinar el sub-flujo de bits basándose al menos en parte en la primera lista y la segunda lista.

- 20 2. El procedimiento según la reivindicación 1, en el que:

la primera lista incluye identificadores de visualización que identifican las visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos,

25 la segunda lista incluye identificadores de visualización que identifican las visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos, y

30 la primera lista incluye uno o más identificadores de visualización que son diferentes a los identificadores de visualización en la lista de visualizaciones de objetivos de profundidad.

3. El procedimiento según la reivindicación 1, en el que:

35 un conjunto de parámetros de secuencia, SPS, especifica las posibles dependencias entre las visualizaciones y los componentes de visualización aplicables al SPS;

determinar la primera lista comprende determinar, basándose al menos en parte en la pluralidad de visualizaciones de objetivos y las posibles dependencias especificadas por el SPS, la primera lista; y

40 determinar la segunda lista comprende determinar, basándose al menos en parte en la pluralidad de visualizaciones de objetivos y las posibles dependencias especificadas por el SPS, la segunda lista.

4. El procedimiento según la reivindicación 1, en el que el flujo de bits 3DV comprende una serie unidades de capa de abstracción de red, NAL, y la determinación del sub-flujo de bits comprende determinar, basándose al menos en parte en si la primera lista o la segunda lista especifica un identificador de visualización de una unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.

5. El procedimiento según la reivindicación 4, en el que determinar la primera lista y la segunda lista comprende:

50 determinar una lista de visualizaciones de objetivos de textura de anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para la descodificación de imágenes de anclaje en la pluralidad de visualizaciones de objetivos;

55 determinar una lista de visualizaciones de objetivos de textura sin anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes sin anclaje en la pluralidad de visualizaciones de objetivos;

60 determinar una lista de visualizaciones de objetivos de profundidad de anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar las imágenes de anclaje en la pluralidad de visualizaciones de objetivos; y

determinar una lista de visualizaciones de objetivos de profundidad sin anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar las imágenes sin anclaje en la pluralidad de visualizaciones de objetivos.

6. El procedimiento según la reivindicación 5, en el que determinar si se elimina la unidad NAL del flujo de bits 3DV comprende:

5 determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece a un tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura de anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV, en el que las unidades NAL que pertenecen al tipo de unidad NAL de componente de visualización de profundidad encapsulan extensiones de fragmentos codificados para componentes de visualización de profundidad; y

10 determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura sin anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.

- 15 7. El procedimiento según la reivindicación 6, en el que determinar si se elimina la unidad NAL del flujo de bits 3DV comprende:

20 determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL no es igual a 21, y un elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de textura de anclaje; y

25 determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico del tipo de unidad NAL en la cabecera de la unidad NAL no es igual a 21, y el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de textura sin anclaje.

- 30 8. El procedimiento según la reivindicación 5, en el que determinar si se elimina la unidad NAL del flujo de bits 3DV comprende:

35 determinar, basándose al menos en parte, en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece a un tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de profundidad de anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV, en el que las unidades NAL que pertenecen al tipo de unidad NAL de componente de visualización de profundidad encapsulan extensiones de fragmentos codificados para componentes de visualización de profundidad; y

40 determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de profundidad sin anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.

- 45 9. El procedimiento según la reivindicación 8, en el que determinar si se elimina la unidad NAL del flujo de bits 3DV comprende:

50 determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando un elemento sintáctico de indicador de imagen de anclaje en una cabecera de la unidad NAL es igual a 1, un elemento sintáctico de tipo de unidad NAL en la cabecera de la unidad NAL es igual a 21, y un elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad de anclaje; y

55 determinar que la unidad NAL debe eliminarse del flujo de bits 3DV cuando el elemento sintáctico de indicador de imagen de anclaje en la cabecera de la unidad NAL es igual a 0, el elemento sintáctico del tipo de unidad NAL en la cabecera de la unidad NAL es igual a 21, y el elemento sintáctico de identificador de visualización en la cabecera de la unidad NAL es igual a un identificador de visualización de una visualización en la lista de objetivos de profundidad sin anclaje.

- 60 10. El procedimiento según la reivindicación 1, en el que el procedimiento se realiza mediante un dispositivo de red de suministro de contenido, CDN.

- 65 11. Un dispositivo que comprende:

- 5 medios para determinar una primera lista que indica visualizaciones en un flujo de bits de vídeo tridimensional, 3DV, con las visualizaciones indicadas por la primera lista que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos, con la pluralidad de visualizaciones de objetivos que comprende un subconjunto de las visualizaciones en el flujo de bits 3DV que se pueden descodificar desde el sub-flujo de bits, con el flujo de bits 3DV que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados;
- 10 medios para determinar una segunda lista independiente que indica visualizaciones en el flujo de bits 3DV, con las visualizaciones indicadas por la segunda lista que tienen componentes de visualización de profundidad que son necesarios para descodificar imágenes en la pluralidad de visualizaciones de objetivos, con la primera lista y la segunda lista que indican subconjuntos diferentes de las visualizaciones en el flujo de bits 3DV; y
- 15 medios para determinar el sub-flujo de bits basándose al menos en parte en la primera lista y la segunda lista.
12. El dispositivo de la reivindicación 11, en el que el flujo de bits 3DV comprende una serie de unidades de capa de abstracción de red, NAL, y los medios para determinar el sub-flujo de bits comprenden medios para determinar, basándose al menos en parte en si la primera lista o la segunda lista especifica un identificador de visualización de una unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.
- 20
13. El dispositivo según la reivindicación 12, en el que los medios para determinar la primera lista comprenden:
- 25 medios para determinar una lista de visualizaciones de objetivos de textura de anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para la descodificación de imágenes de anclaje en la pluralidad de visualizaciones de objetivos;
- 30 medios para determinar una lista de visualizaciones de objetivos de textura sin anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de textura que son necesarios para descodificar imágenes sin anclaje en la pluralidad de visualizaciones de objetivos;
- 35 medios para determinar una lista de visualizaciones de objetivos de profundidad de anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para descodificar las imágenes de anclaje en la pluralidad de visualizaciones de objetivos; y
- 40 medios para determinar una lista de visualizaciones de objetivos de profundidad sin anclaje que indica visualizaciones en el flujo de bits 3DV que tienen componentes de visualización de profundidad que son necesarios para la descodificación de las imágenes sin anclaje en la pluralidad de visualizaciones de objetivos.
14. El dispositivo según la reivindicación 13, en el que los medios para determinar si eliminar la unidad NAL del flujo de bits 3DV comprenden:
- 45 medios para determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece a un tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura de anclaje especifica el identificador de visualización de la unidad NAL, si se debe eliminar la unidad NAL del flujo de bits 3DV, en el que las unidades NAL que pertenecen al tipo de unidad NAL de componente de visualización de profundidad encapsulan extensiones de fragmentos codificados para componentes de visualización de profundidad; y
- 50 medios para determinar, basándose al menos en parte en si la unidad NAL pertenece a una imagen de anclaje, si la unidad NAL pertenece al tipo de unidad NAL de componente de visualización de profundidad, y si la lista de visualizaciones de objetivos de textura sin anclaje especifica el identificador de visualización de la unidad NAL, si eliminar la unidad NAL del flujo de bits 3DV.
- 55
15. Un medio de almacenamiento legible por ordenador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores de un dispositivo, configuran el dispositivo para:
- 60 determinar una primera lista que indique visualizaciones en un flujo de bits 3DV, con las visualizaciones indicadas por la primera lista que tienen componentes de visualización de textura que son necesarios para descodificar imágenes en una pluralidad de visualizaciones de objetivos, con la pluralidad de visualizaciones de objetivos que comprende un subconjunto de las visualizaciones en el flujo de bits 3DV que se puede descodificar a partir del sub-flujo de bits, con el flujo de bits 3DV que incluye componentes de visualización de textura codificados y componentes de visualización de profundidad codificados;
- 65

determinar una segunda lista independiente que indique visualizaciones en el flujo de bits 3DV, con las visualizaciones indicadas por la segunda lista que tienen componentes de visualización de profundidad que son necesarios para decodificar imágenes en la pluralidad de visualizaciones de objetivos, con la primera lista y la segunda lista que indican diferentes subconjuntos de las visualizaciones en el flujo de bits 3DV; y

5

determinar el sub-flujo de bits basándose al menos en parte en la primera lista y la segunda lista.

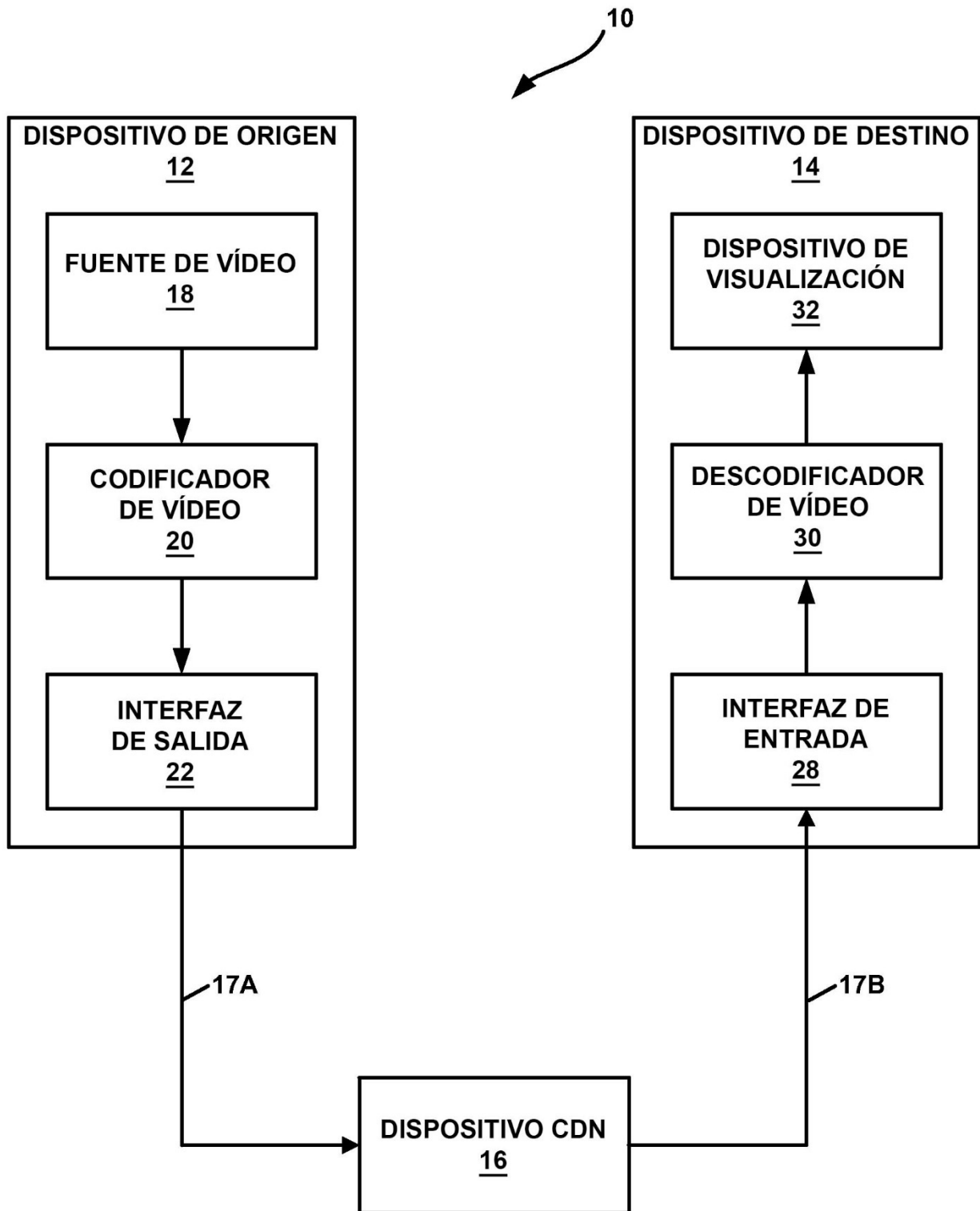


FIG. 1

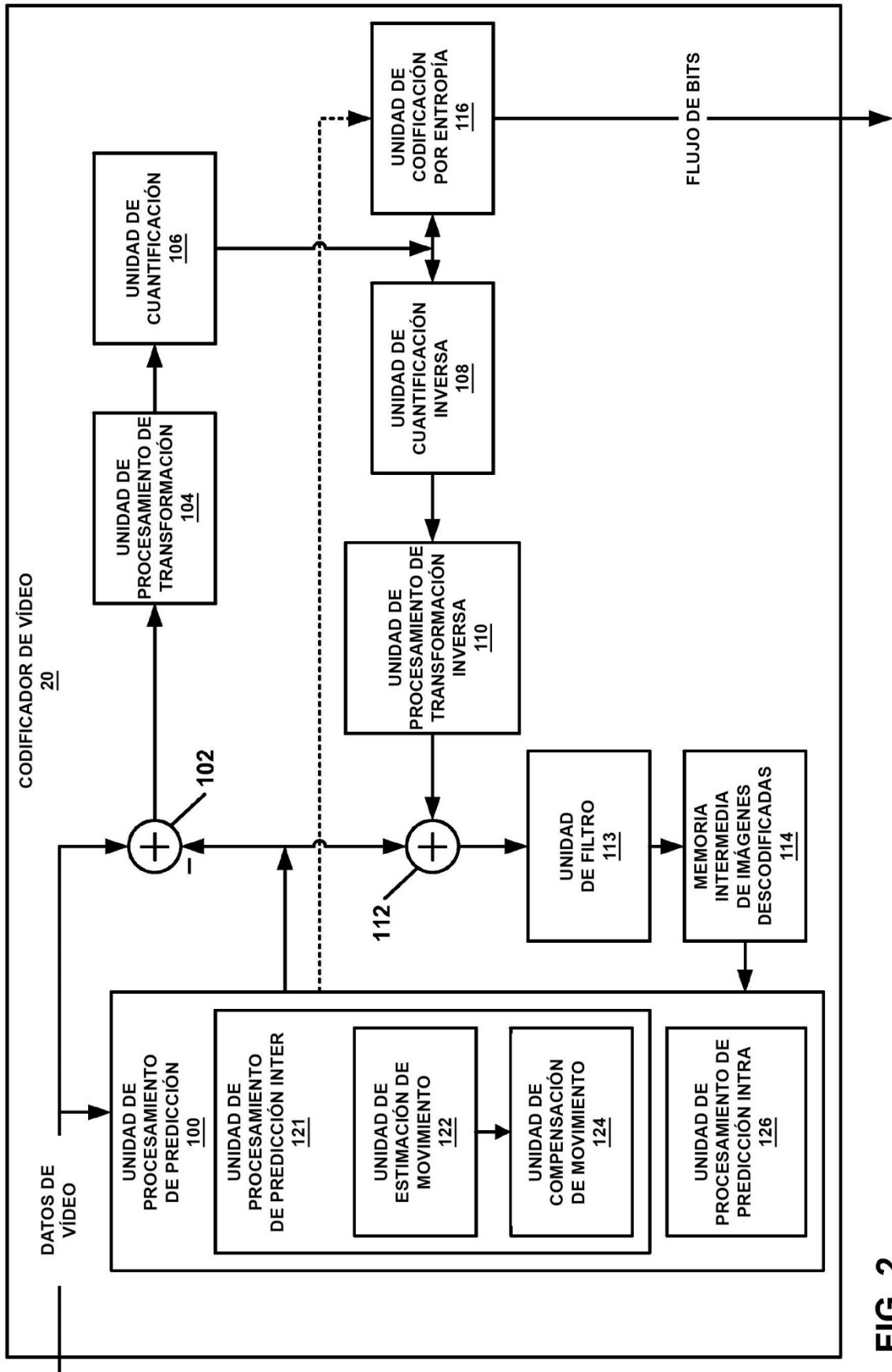


FIG. 2

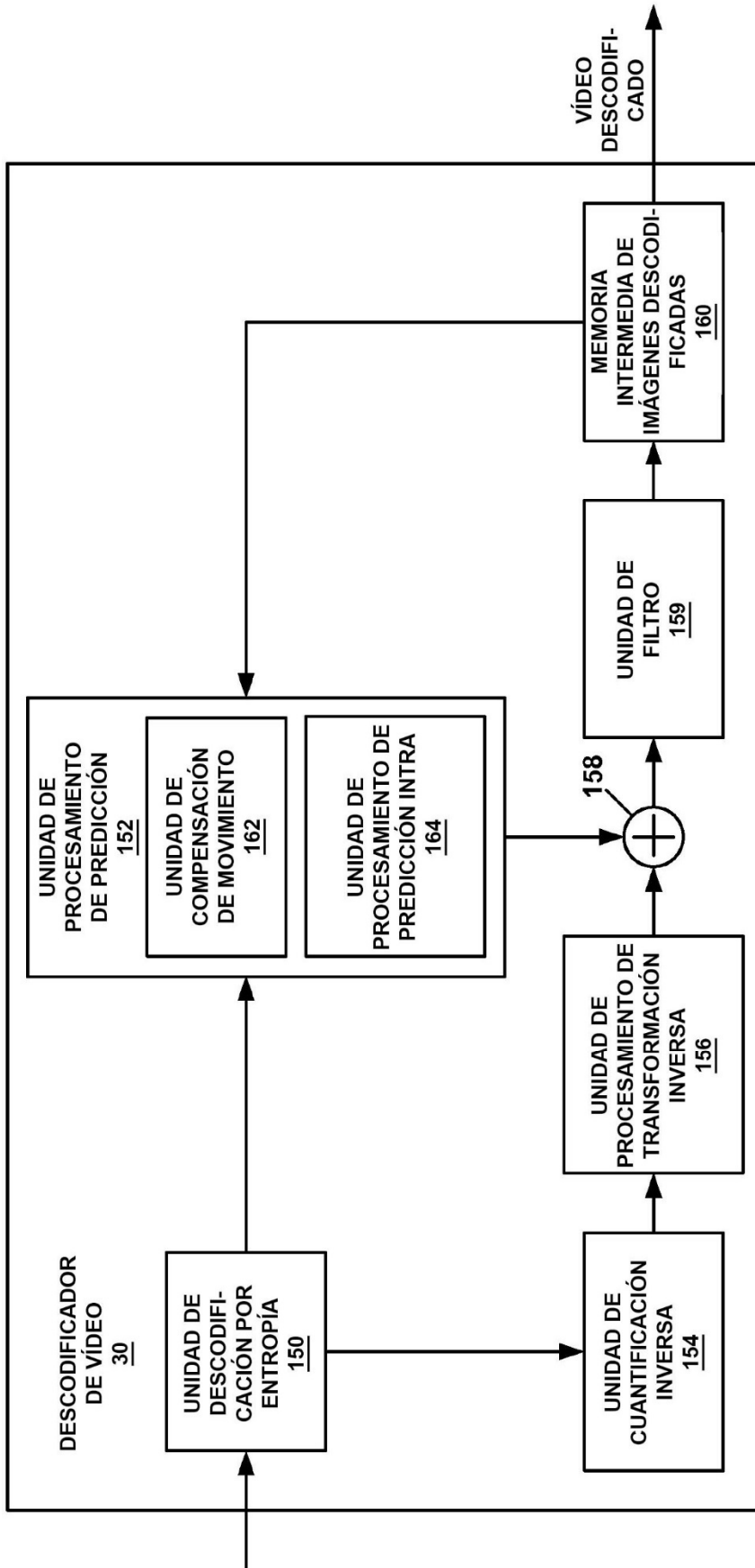


FIG. 3

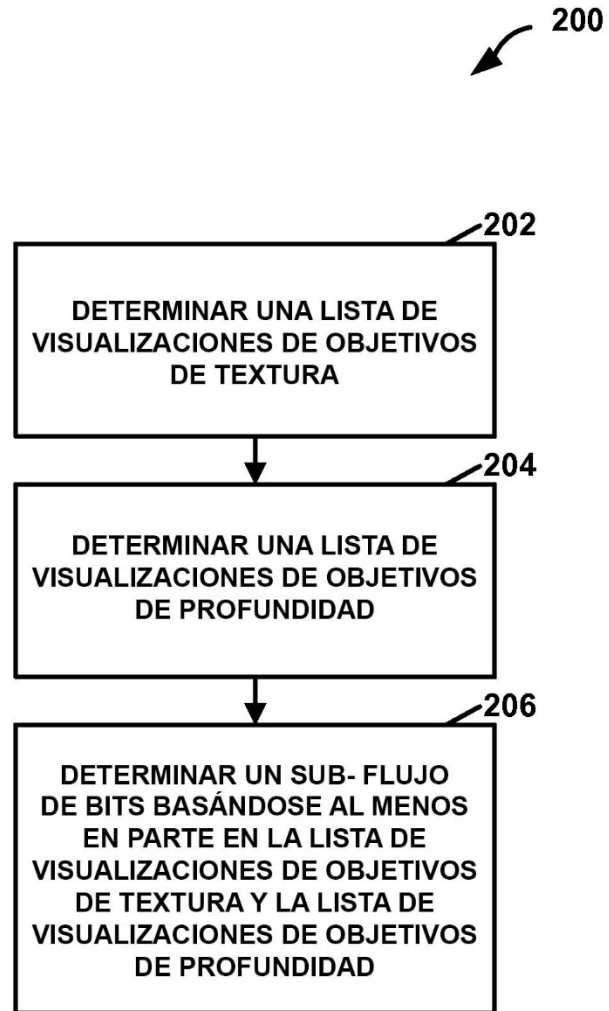


FIG. 4

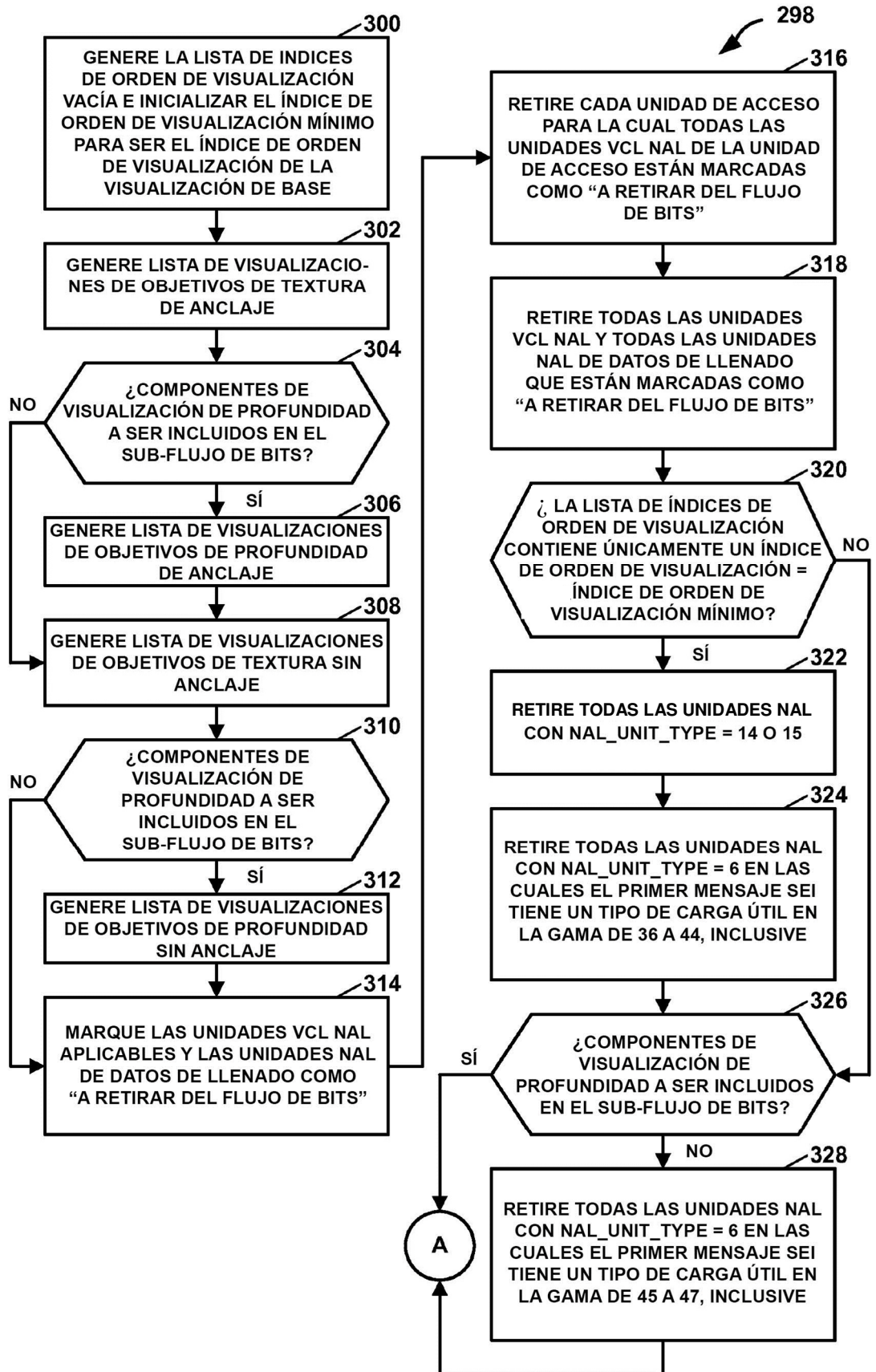


FIG. 5

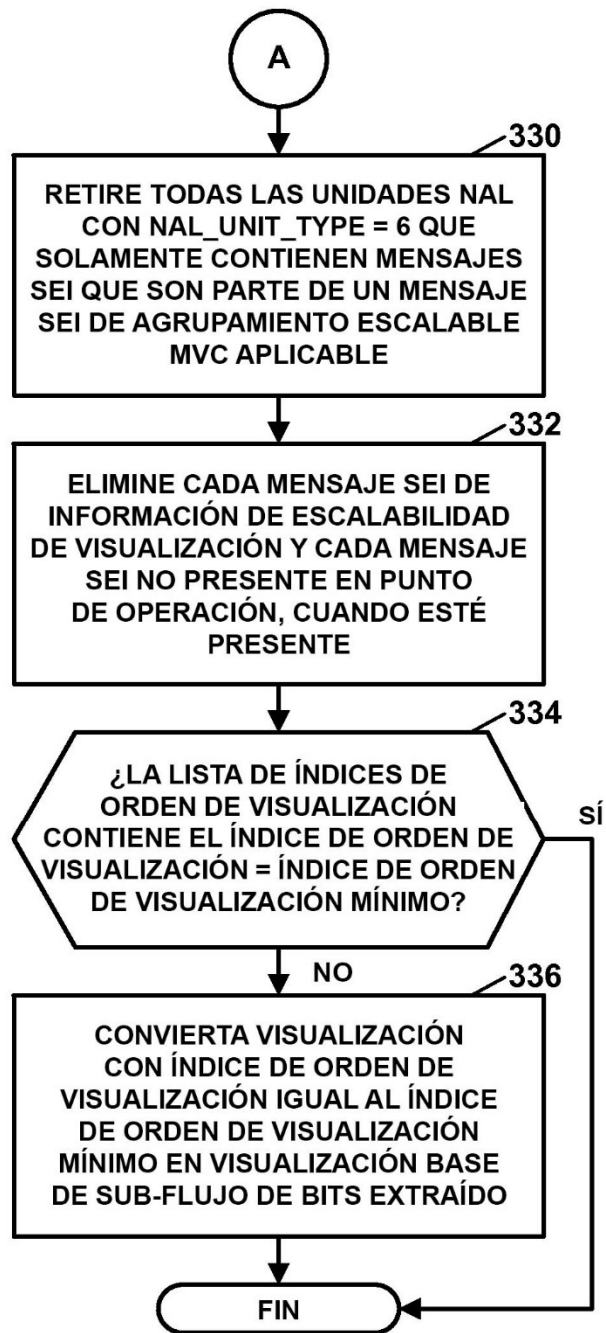


FIG. 6

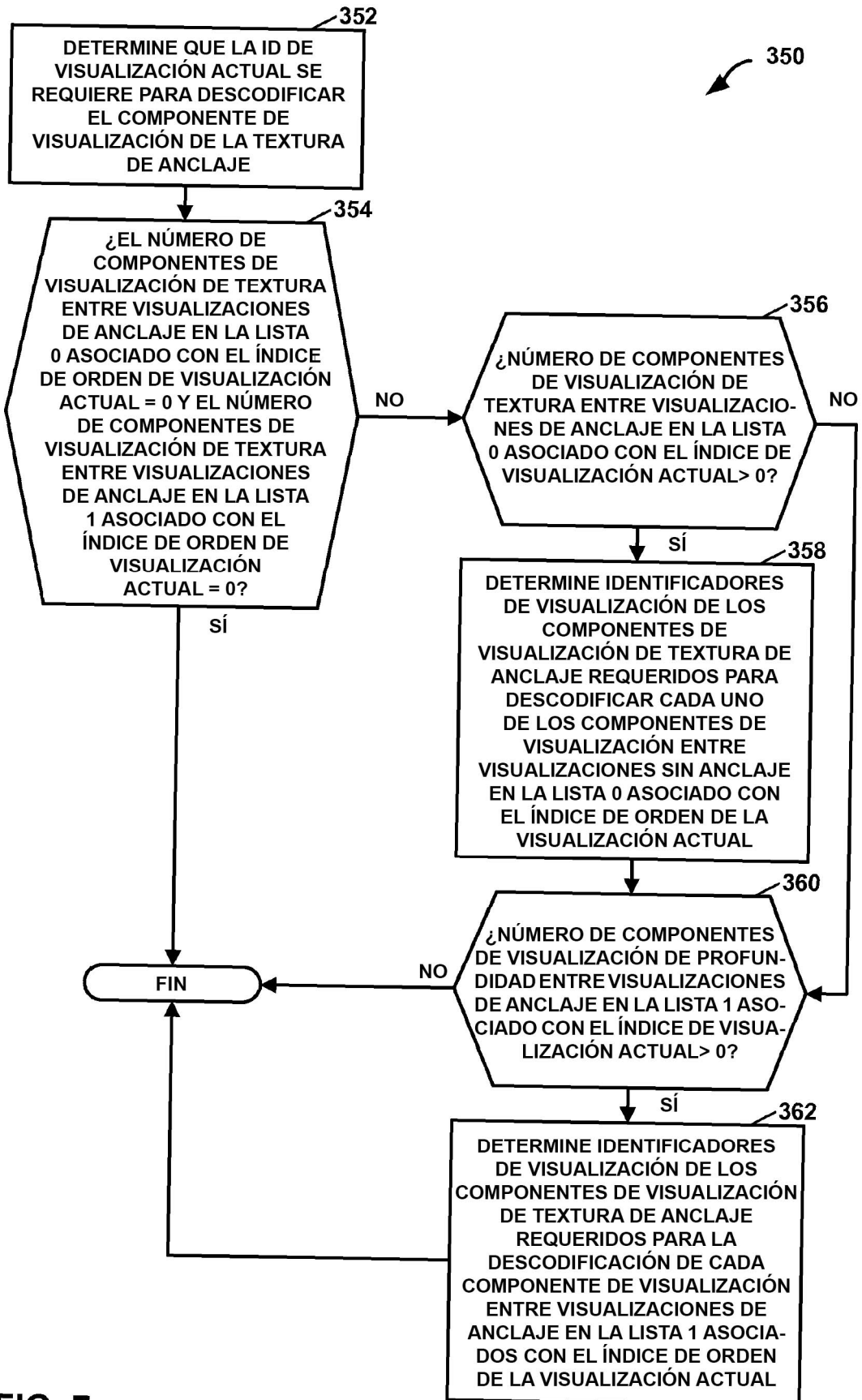


FIG. 7

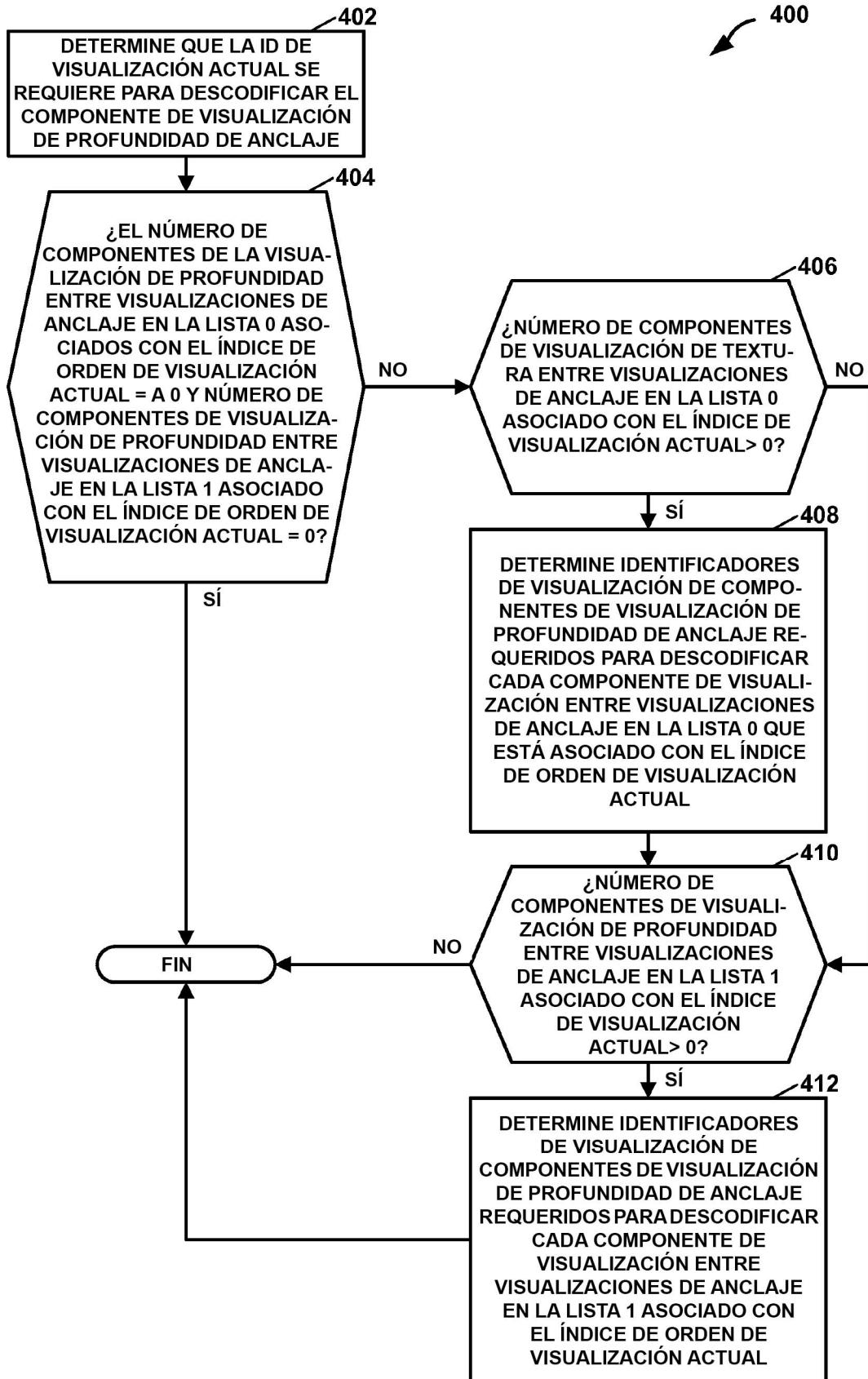


FIG. 8

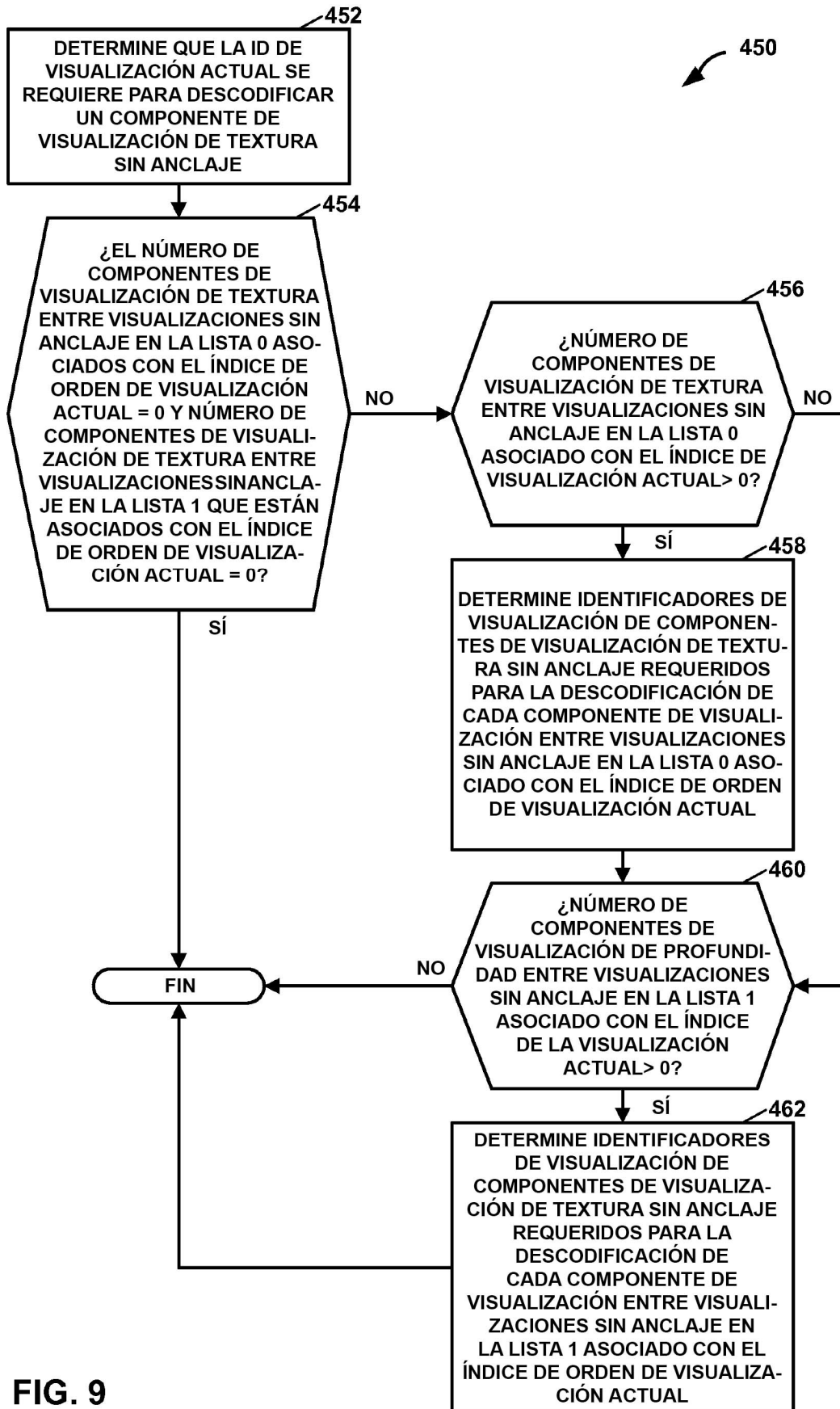


FIG. 9

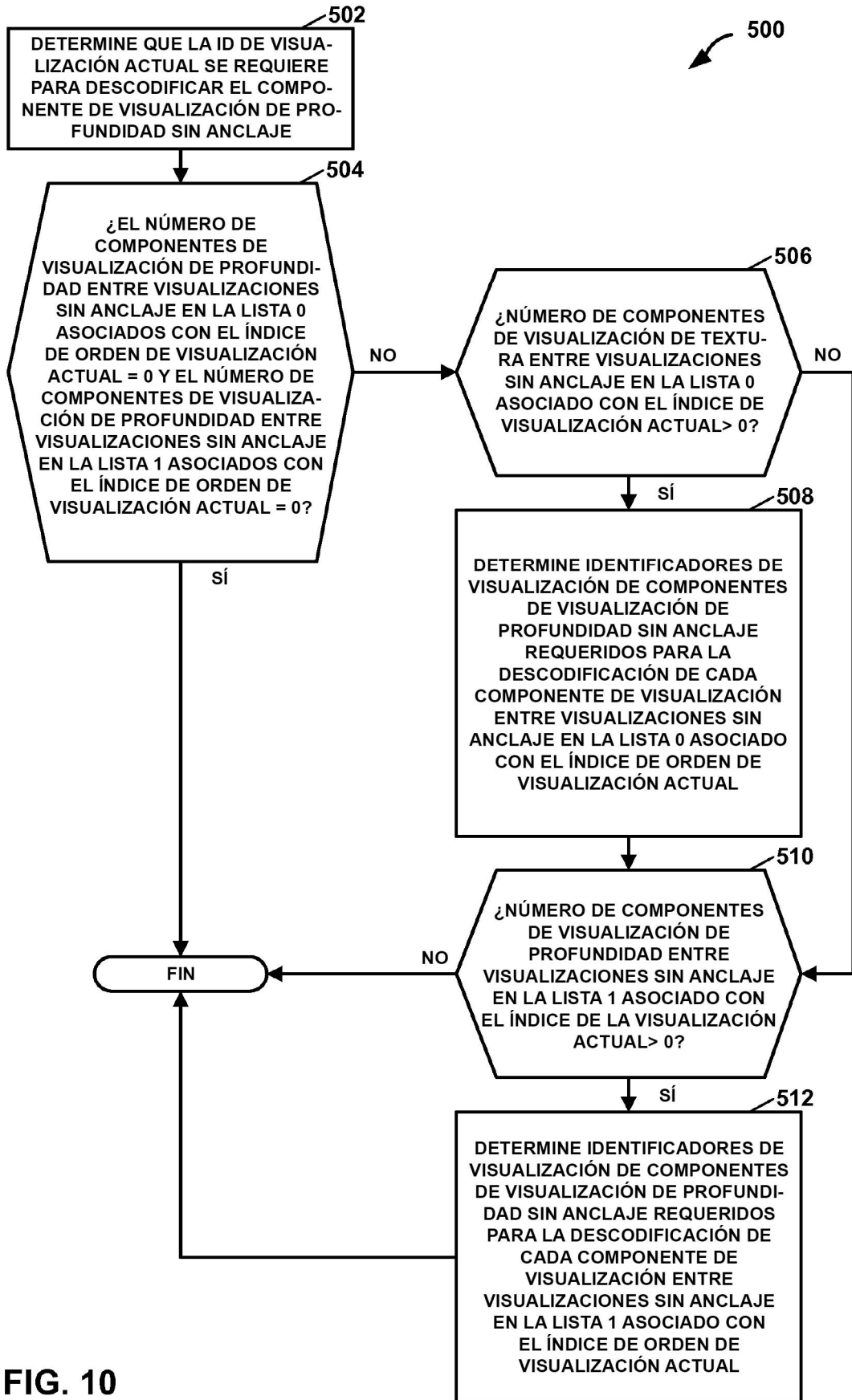


FIG. 10

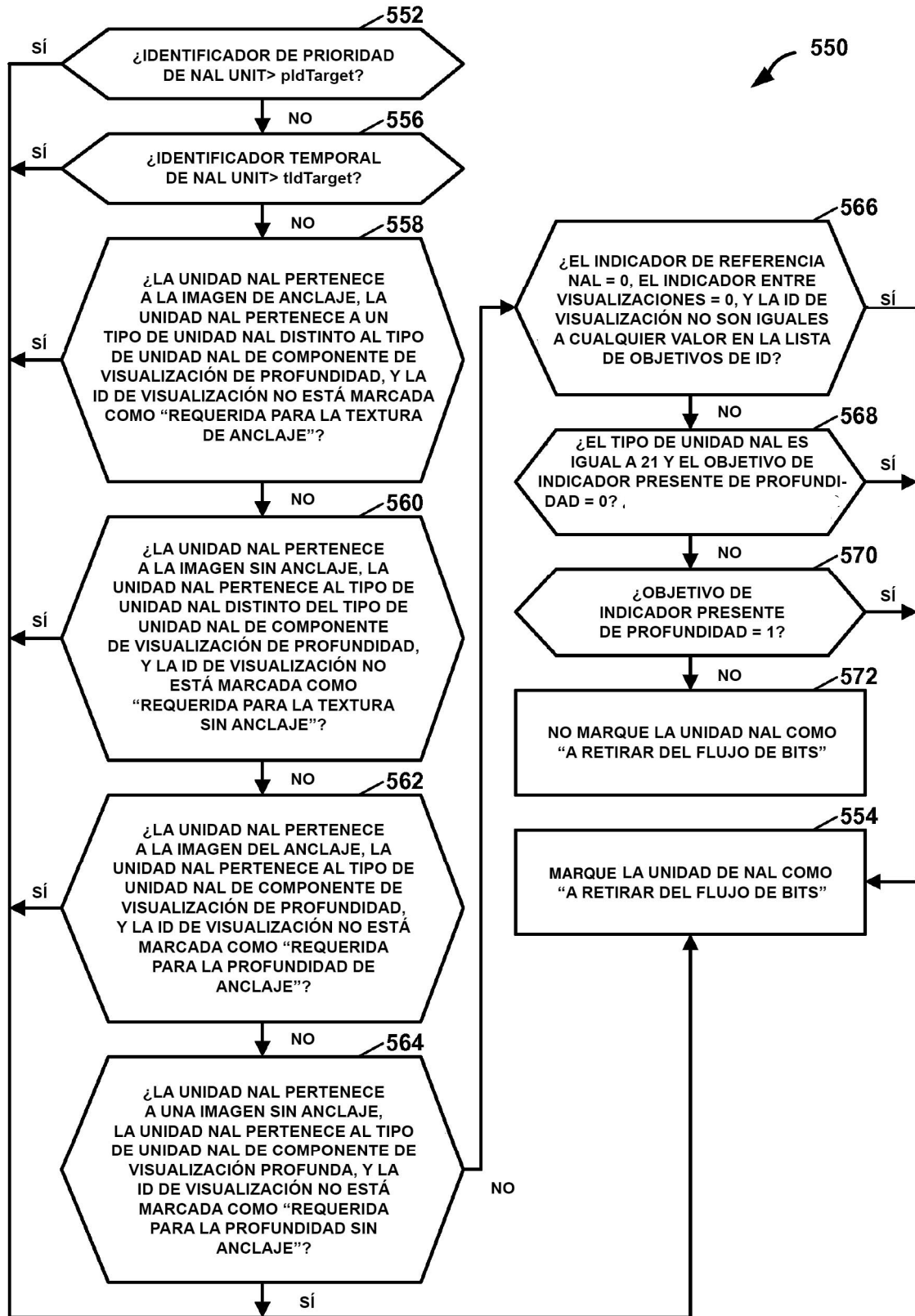


FIG. 11

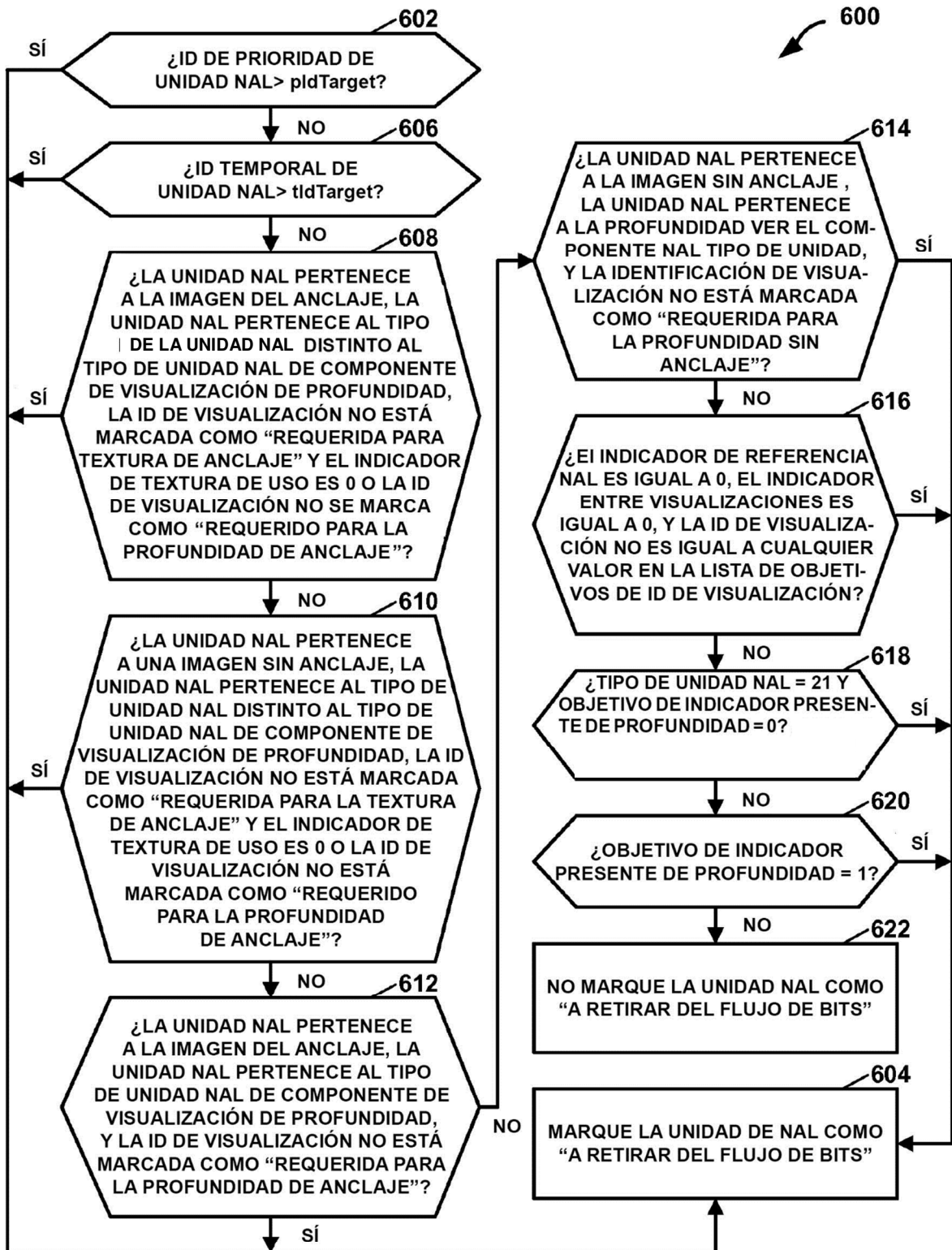


FIG. 12

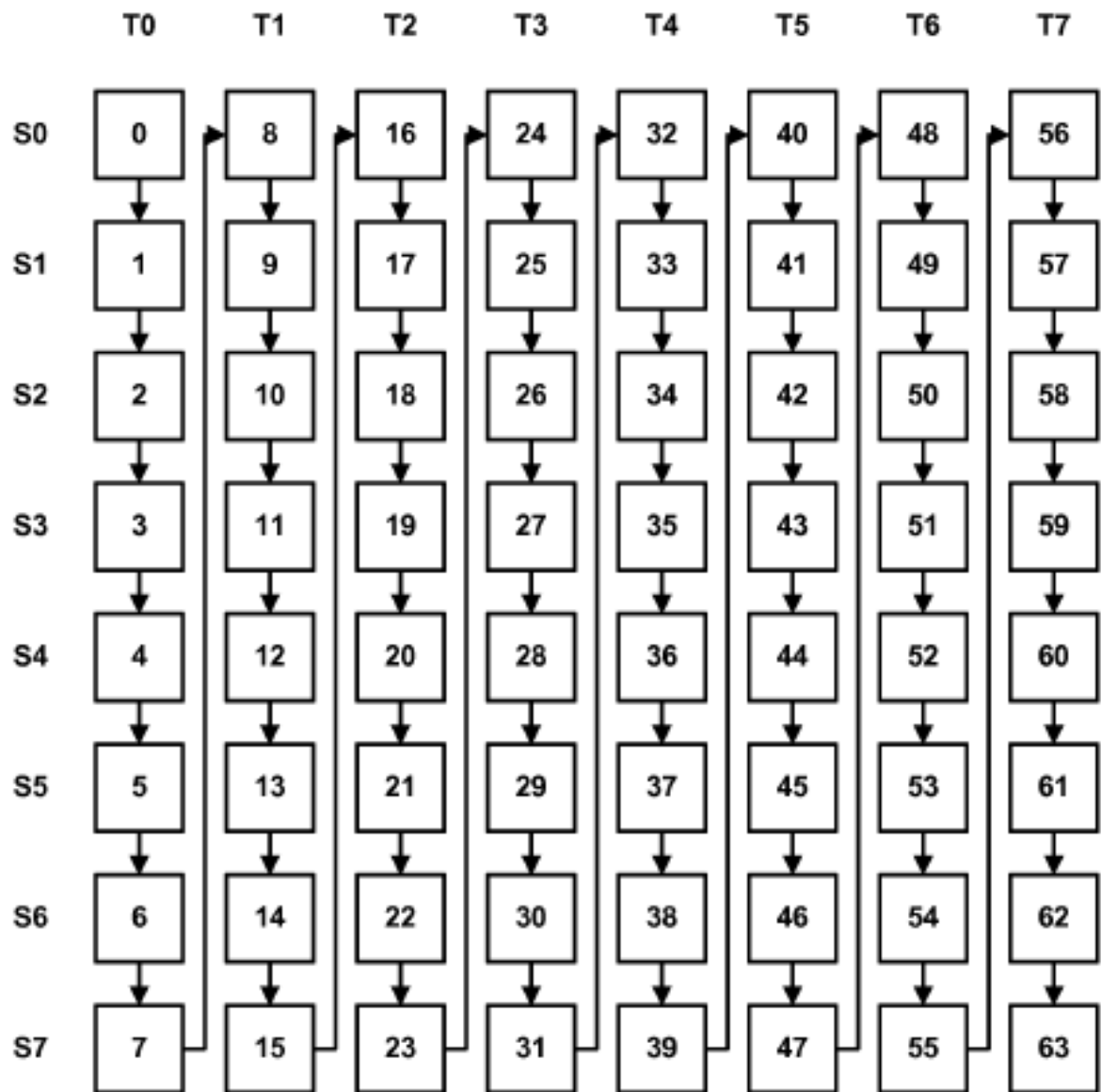


FIG. 13

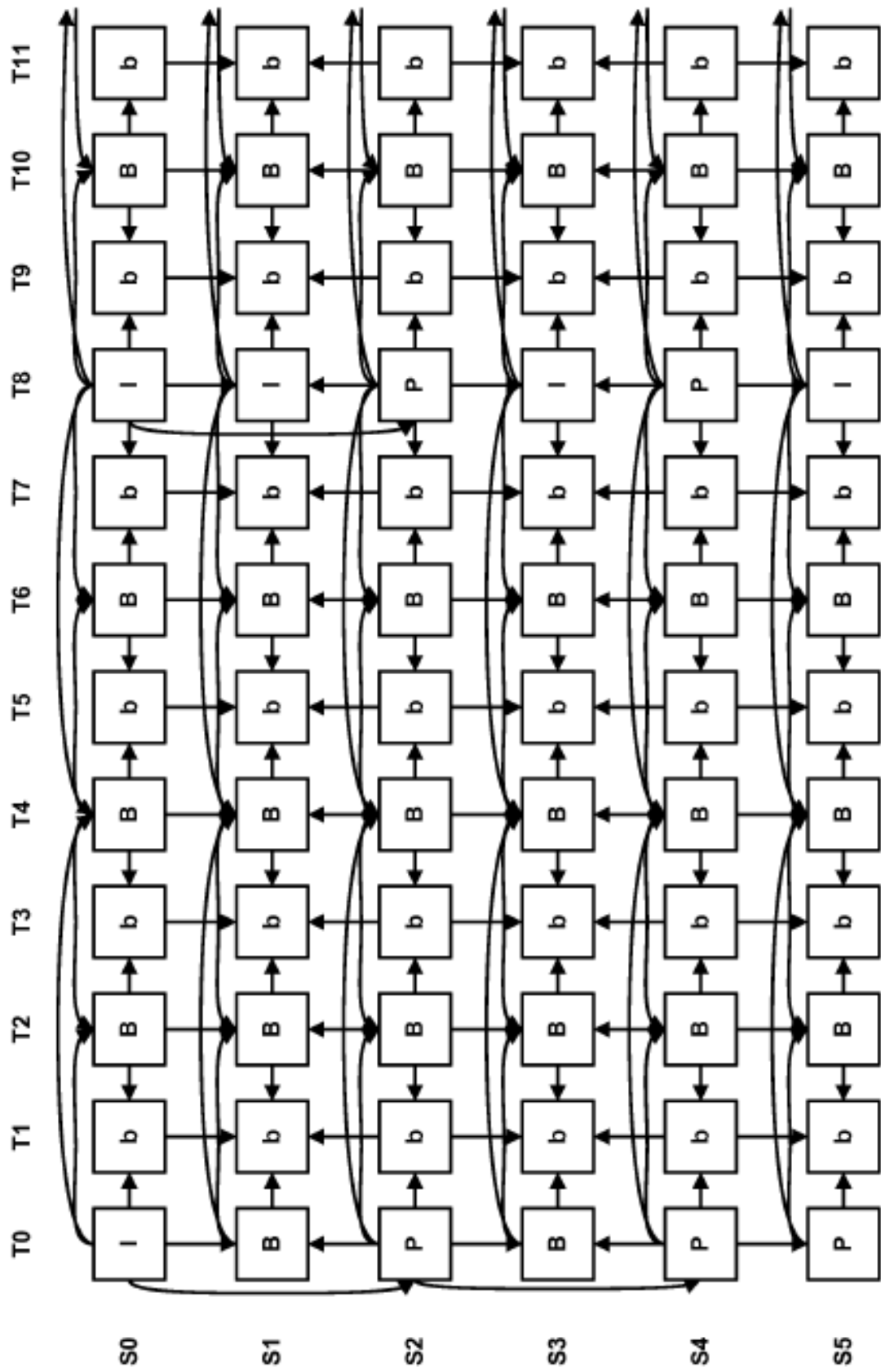


FIG. 14