

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 656 908**

51 Int. Cl.:

H04N 19/17 (2014.01)
H04N 19/33 (2014.01)
H04N 19/70 (2014.01)
H04N 19/50 (2014.01)
H04N 19/436 (2014.01)
H04N 19/82 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **05.03.2014** **PCT/US2014/020711**
87 Fecha y número de publicación internacional: **12.09.2014** **WO14138217**
96 Fecha de presentación y número de la solicitud europea: **05.03.2014** **E 14714009 (9)**
97 Fecha y número de publicación de la concesión europea: **01.11.2017** **EP 2965517**

54 Título: **Procesamiento paralelo para codificación de vídeo**

30 Prioridad:

05.03.2013 US 201361773095 P
05.04.2013 US 201361808734 P
03.05.2013 US 201361819494 P
04.03.2014 US 201414196295

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
28.02.2018

73 Titular/es:

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

72 Inventor/es:

RAPAKA, KRISHNAKANTH;
LI, XIANG;
CHEN, JIANLE;
WANG, YE-KUI y
PU, WEI

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 656 908 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

Procesamiento paralelo para codificación de vídeo

5 CAMPO TÉCNICO

[0001] Esta divulgación se refiere a la codificación de vídeo.

ANTECEDENTES

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

25 [0003] Las técnicas de codificación de vídeo incluyen la predicción espacial (intra-imagen) y/o la predicción temporal (inter-imagen) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación de vídeo basándose en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse bloques arbolados, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo de un fragmento intra-codificado (I) de una imagen se codifican mediante la predicción espacial con respecto a las muestras de referencia de los bloques contiguos de la misma imagen. Los bloques de vídeo de un fragmento inter-codificado (P o B) de una imagen pueden usar la predicción espacial con respecto a las muestras de referencia de los bloques contiguos de la misma imagen, o la predicción temporal con respecto a las muestras de referencia de otras imágenes de referencia. Las imágenes pueden denominarse tramas y las imágenes de referencia pueden denominarse tramas de referencia.

35 [0004] La predicción espacial o temporal dan como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se va a codificar y el bloque predictivo. Un bloque inter-codificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intra-codificado se codifica de acuerdo con un modo de intra-codificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de los píxeles al dominio de las transformaciones, dando como resultado unos coeficientes de transformación residual, que posteriormente se pueden cuantificar. Los coeficientes de transformación cuantificados, dispuestos inicialmente en una matriz bidimensional, pueden escanearse con el fin de generar un vector unidimensional de coeficientes de transformación, y puede aplicarse codificación por entropía para lograr aún más compresión. Se hace referencia a los siguientes documentos:

45 "Descripción de la tecnología de codificación de vídeo escalable propuesta por Canon Research Centre France", por Lasserre et al, 102 reunión de MPEG, 15-19 de octubre de 2012, Shanghai n.º M26540;
 "AHG9: indicador de predicción entre capas e indicador de predicción de sintaxis entre capas", 12. Reunión JCT-VC, JCTVC-L0071, 7 de enero de 2013; y
 50 "Descripción de la propuesta de tecnología de codificación de vídeo escalable por Sharp (Propuesta 2)", 102 Reunión de MPEG, 15-19 de octubre de 2012, M26599.

RESUMEN

55 [0005] La invención se define en las reivindicaciones a las que se dirige ahora la referencia.

[0006] En general, esta divulgación describe técnicas para procesamiento paralelo de codificación de vídeo. Por ejemplo, esta divulgación describe técnicas para el procesamiento paralelo en extensiones escalables de la norma de codificación de vídeo de alta eficiencia (HEVC), aunque las técnicas se pueden aplicar a otras normas de codificación de vídeo y extensiones de tales normas, como HEVC multivisión (MV-HEVC). Se describen varias soluciones de ejemplo para realizar de manera eficiente el procesamiento paralelo en capas (que puede incluir capas escalables de varias dimensiones y múltiples vistas), muestreo ascendente, filtrado entre capas (interlayer), predicción de sintaxis entre capas, como predicción de movimiento / modo entre capas, etc., y predicciones de texturas entre capas, tales como textureBI, predicción residual entre capas, y similares, cuando se utiliza más de un elemento (tile) para cada capa, y en el que cada capa puede ser una capa escalable, una vista o una profundidad

65 [0007] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras

características, objetos y ventajas resultarán evidentes a partir de la descripción y los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

- 5 [0008]
- 10 La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar técnicas para procesamiento de datos de vídeo en paralelo.
- 15 La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas para procesamiento de datos de vídeo en paralelo.
- 20 La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas para procesamiento de datos de vídeo en paralelo.
- 25 La FIG. 4 es un diagrama conceptual que ilustra varias dimensiones escalables para la codificación de vídeo escalable (SVC).
- 30 La FIG. 5 es un diagrama conceptual que ilustra un ejemplo de la estructura de codificación SVC.
- 35 La FIG. 6 es un diagrama conceptual que ilustra unidades de acceso (AU) de ejemplo.
- 40 La FIG. 7 es un diagrama conceptual que ilustra elementos de ejemplo de acuerdo con la norma de codificación de vídeo de alta eficiencia (HEVC).
- 45 La FIG. 8 es un diagrama conceptual que ilustra los frentes de onda para el procesamiento paralelo de frentes de onda (WPP).
- 50 La FIG. 9 es un diagrama conceptual que ilustra un ejemplo de procesamiento paralelo en una extensión SVC del decodificador HEVC (SHVC).
- 55 La FIG. 10 muestra un ejemplo de escalabilidad espacial extendida donde algunas muestras de capa de mejora (EL) no tienen muestras de capa base correspondientes.
- 60 La FIG. 11 muestra un ejemplo de escalabilidad espacial extendida donde algunas muestras de capa base (BL) no tienen muestras de capa de mejora correspondientes.
- La FIG. 12 es un diagrama conceptual que ilustra una imagen de capa de referencia y una imagen de capa de mejora.
- La FIG. 13 es un diagrama conceptual que ilustra imágenes de ejemplo que tienen diferentes relaciones de aspecto.
- La FIG. 14 es un diagrama conceptual que ilustra una imagen de capa de mejora y una imagen de capa de referencia que tienen diferentes números de elementos.
- La FIG. 15 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la habilitación o inhabilitación de predicción entre capas de acuerdo con las técnicas de esta divulgación.
- La FIG. 16 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes de una capa de mejora y una capa de referencia en paralelo cuando las imágenes tienen relaciones de aspecto diferentes, de acuerdo con las técnicas de esta divulgación.
- La FIG. 17 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar una imagen de capa de mejora basándose en si los límites de los elementos de la imagen de capa de mejora están alineados con los límites de los elementos de una imagen de capa de referencia.
- La FIG. 18 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar elementos de una imagen de capa de mejora en paralelo con una imagen de capa de referencia, incluso cuando los límites de los elementos no están alineados, de acuerdo con las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

- 65 [0009] Entre las normas de codificación de vídeo se incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC

MPEG-4 AVC), incluyendo sus extensiones de codificación de vídeo ajustable a escala (SVC) y de codificación de vídeo multivista (MVC). Un borrador conjunto de SVC y MVC se describe en "Codificación de vídeo avanzada para servicios audiovisuales genéricos", recomendación ITU-T H.264, marzo de 2010.

[0010] Además, existe una nueva norma de codificación de vídeo, concretamente la Codificación de Vídeo de Alta Eficiencia (HEVC) que está siendo desarrollada por el Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC. HEVC, también conocido como ITU-T H.265, se describe en ITU-T H.265, SERIE H: SISTEMAS AUDIOVISUALES Y MULTIMEDIA, Infraestructura de Servicios Audiovisuales-Codificación de Vídeo en Movimiento, "Codificación de Vídeo de Alta Eficiencia", abril de 2013, disponible en <http://www.itu.int/rec/T-REC-H.265-201304-l/en>.

[0011] Ciertos detalles sobre SVC, con respecto tanto a la extensión SVC de H.264/AVC como a la extensión SVC de HEVC, se explican a continuación con respecto a las FIGs. 4-9.

[0012] Las soluciones existentes para el procesamiento paralelo en contextos de codificación de múltiples capas se pueden asociar con al menos los siguientes problemas:

[0013] En las extensiones de HEVC, puede haber más interdependencias relacionadas con diferentes capas de la misma unidad de acceso o diferentes unidades de acceso. Por ejemplo, la descodificación de una imagen de capa de mejora puede depender de una imagen de capa de referencia descodificada. La extensión SVC de HEVC (SHVC) emplea un marco de descodificación escalable de múltiples bucles. En el marco actual de SHVC, todas las imágenes de capa de referencia de las que depende una imagen de capa actual (a través de cualquier tipo de predicción entre capas) tienen que descodificarse para comenzar a descodificar la imagen de capa actual. Esto puede aumentar el retardo general del códec en ciertas arquitecturas de procesamiento paralelo para algunos sistemas de descodificación.

[0014] Además, cuando se realiza procesamiento paralelo de diferentes capas con más de un elemento en cada capa, el muestreo ascendente, el filtrado entre capas o la predicción de sintaxis entre capas para algunas regiones pueden cruzar los límites de los elementos en la imagen de capa de referencia. Debido al procesamiento paralelo con elementos, no se puede garantizar la disponibilidad de muestras reconstruidas en la imagen de capa de referencia a lo largo de los límites de los elementos, por lo que no se puede realizar correctamente el muestreo ascendente o el filtrado entre capas. En consecuencia, el muestreo ascendente o el filtrado entre capas debe posponerse hasta que se descodifique toda la imagen de capa de referencia o se interrumpa parcialmente el procesamiento paralelo de diferentes capas.

[0015] Además, la especificación de la alineación de límites de los elementos descrita en K. Sühring, et al., Alineación de límites de los elementos y restricciones de predicción entre capas para SHVC y MV-HEVC, JCTVC-M0464, abril de 2013 (en adelante "JCTVC-M0464") está asociada con al menos los siguientes problemas. Para los escenarios de casos de uso de escalabilidad de relación de aspecto o escalabilidad espacial extendida, una muestra de capa de mejora puede no tener una muestra de capa base correspondiente y viceversa. Por lo tanto, la especificación de la alineación de los límites de los elementos como se describe en F. Henry y S. Pateux, "Procesamiento paralelo de frente de onda", JCTVC-E196, marzo de 2011(en adelante, "JCTVC-E196") puede rechazar por completo el uso de elementos a través de las capas. La FIG. 10 muestra un ejemplo de escenario de escalabilidad espacial extendida donde una muestra de capa de mejora no tiene una muestra de capa base correspondiente. La FIG. 11 muestra un ejemplo de un escenario de escalabilidad espacial extendida donde una muestra de capa base no tiene una muestra de capa de mejora correspondiente. JCTVC-M0464 y JCTVC-E196 se incorporan aquí como referencia en su totalidad.

[0016] La FIG. 10 muestra un ejemplo de escalabilidad espacial extendida donde algunas muestras de EL (capa de mejora) no tienen muestras de capa base correspondientes. La FIG. 11 muestra un ejemplo de escalabilidad espacial extendida donde algunas muestras BL (capa base) no tienen muestras de capa de mejora correspondientes.

[0017] Esta divulgación describe dos categorías de técnicas que pueden utilizarse para soportar el procesamiento paralelo, por ejemplo, en las extensiones de HEVC. Las técnicas de estas dos categorías se pueden usar solas o independientemente. Las técnicas pueden aplicarse a codificación de vídeo escalable, codificación de vídeo de múltiples vistas (con o sin profundidad) y otras extensiones a HEVC y otros códecs de vídeo. La primera categoría está dirigida a la predicción entre capas restringida. La segunda categoría está dirigida a la indicación. Ejemplos de técnicas de ambas categorías se explican con mayor detalle a continuación.

[0018] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y descodificación de vídeo 10 de ejemplo que puede utilizar técnicas para procesamiento de datos de vídeo en paralelo. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados, a descodificar en un momento posterior mediante un dispositivo de destino 14. En particular, el dispositivo de origen 12 proporciona los datos de vídeo al dispositivo de destino 14 mediante un medio legible por ordenador 16. El dispositivo de origen 12 y

el dispositivo de destino 14 pueden comprender cualquiera de entre una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, notebook (es decir, portátiles), ordenadores de tableta, descodificadores, equipos telefónicos inalámbricos tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, un dispositivo de transmisión de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0019] El dispositivo de destino 14 puede recibir los datos de vídeo codificados para ser descodificados mediante el medio legible por ordenador 16. El medio legible por ordenador 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el medio legible por ordenador 16 puede comprender un medio de comunicación que habilita al dispositivo fuente 12 para transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0020] En algunos ejemplos, pueden emitirse datos codificados desde la interfaz de salida 22 hasta un dispositivo de almacenamiento. De forma similar, se puede acceder a los datos codificados del dispositivo de almacenamiento mediante una interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera entre una diversidad de medios de almacenamiento de datos, de acceso distribuido o local, tales como una unidad de disco rígido, discos Blu-ray, discos DVD, discos CD-ROM, memoria flash, memoria volátil o no volátil, u otros medios adecuados cualesquiera de almacenamiento digital, para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de almacenamiento puede corresponder a un servidor de ficheros o a otro dispositivo de almacenamiento intermedio que pueda almacenar los datos de vídeo codificados, generados por el dispositivo de origen 12.

[0021] El dispositivo de destino 14 puede acceder a datos de vídeo almacenados desde el dispositivo de almacenamiento mediante flujo o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Los ejemplos de servidores de archivos incluyen un servidor de la red (por ejemplo, para una página web), un servidor de FTP, dispositivos de almacenamiento anexos a la red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de cualquier conexión de datos estándar, incluyendo una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión cableada (por ejemplo, DSL, módem de cable, etc.), o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento puede ser una transmisión por flujo, una transmisión de descarga o una combinación de ambas.

[0022] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, en soporte de cualquiera entre una diversidad de aplicaciones de multimedios, tales como radiodifusiones de televisión por el aire, transmisiones de televisión por cable, radiodifusiones de televisión por satélite, transmisiones de vídeo por flujo de Internet, tales como el flujo adaptativo dinámico sobre HTTP (DASH), el vídeo digital que se codifica en un medio de almacenamiento de datos, la descodificación de vídeo digital almacenado en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema 10 puede configurarse para soportar la transmisión de vídeo unidireccional o bidireccional a fin de prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0023] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. El dispositivo de destino 14 incluye una interfaz de entrada 28, un descodificador de vídeo 30 y un dispositivo de visualización 32. De acuerdo con esta divulgación, el codificador de vídeo 20 del dispositivo de origen 12 puede estar configurado para aplicar las técnicas para procesamiento de datos de vídeo en paralelo. En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde una fuente de vídeo externo 18, tal como una cámara externa. Asimismo, el dispositivo de destino 14 puede mantener interfaces con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0024] El sistema ilustrado 10 de la FIG. 1 es simplemente un ejemplo. Las técnicas para procesamiento de datos de vídeo en paralelo pueden realizarse mediante cualquier dispositivo de codificación y/o descodificación de vídeo digital. Aunque, en general, las técnicas de esta divulgación se llevan a cabo mediante un dispositivo de codificación de vídeo, las técnicas también pueden llevarse a cabo mediante un codificador/descodificador de vídeo, denominado típicamente "CÓDEC". Además, las técnicas de esta divulgación también pueden llevarse a cabo mediante un preprocesador de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 son simplemente ejemplos de tales

dispositivos de codificación, donde el dispositivo de origen 12 genera datos de vídeo codificados para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de manera esencialmente simétrica, de modo que cada uno de los dispositivos 12, 14 incluya componentes de codificación y de decodificación de vídeo. Por lo tanto, el sistema 10 puede dar soporte a una transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo, para la transmisión por flujo de vídeo, la reproducción de vídeo, la radiodifusión de vídeo o la videoteléfono.

[0025] La fuente de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado previamente y/o una interfaz de alimentación de vídeo para recibir vídeo de un proveedor de contenidos de vídeo. Como una alternativa adicional, la fuente de vídeo 18 puede generar datos, basados en gráficos de ordenador, como el vídeo de origen, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, si la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados teléfonos con cámara o videoteléfonos. Sin embargo, como se ha mencionado anteriormente, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el vídeo capturado, pre-capturado o generado por ordenador puede ser codificado por el codificador de vídeo 20. La información de vídeo codificada puede ser entonces emitida por la interfaz de salida 22 sobre un medio legible por ordenador 16.

[0026] El medio legible por ordenador 16 puede incluir medios transitorios, tales como una radiodifusión inalámbrica o una transmisión de red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorio), tales como un disco duro, una unidad de memoria flash, un disco compacto, un disco de vídeo digital, un disco Blu-ray u otros medios legibles por ordenador. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y proporcionar los datos de vídeo codificados al dispositivo de destino 14, por ejemplo, mediante transmisión por red. De manera similar, un dispositivo informático de una instalación de producción de un medio, tal como una instalación de grabación de discos, puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y producir un disco que contiene los datos de vídeo codificados. Por lo tanto, puede entenderse que el medio legible por ordenador 16 incluye uno o más medios legibles por ordenador de varias formas, en varios ejemplos.

[0027] La interfaz de entrada 28 del dispositivo de destino 14 recibe información desde el medio legible por ordenador 16. La información del medio legible por ordenador 16 puede incluir información sintáctica definida por el codificador de vídeo 20, que también es usada por el decodificador de vídeo 30, que incluye elementos sintácticos que describen características y/o el procesamiento de bloques y otras unidades codificadas, por ejemplo, grupos de imágenes (GOP). El dispositivo de visualización 32 muestra los datos de vídeo descodificados a un usuario y puede comprender cualquiera entre una variedad de dispositivos de visualización, tales como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0028] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con una norma de codificación de vídeo, tal como la norma de Codificación de Vídeo de Alta Eficencia (HEVC), actualmente en fase de desarrollo, y pueden ajustarse al modelo de prueba de la HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con otras normas privadas o industriales, tales como la norma ITU-T H.264, de forma alternativa denominada MPEG-4, Parte 10, codificación de vídeo avanzada (AVC), o extensiones de dichas normas. Sin embargo, las técnicas de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de codificación de vídeo incluyen las de MPEG-2 e ITU-T H.263. Aunque no se muestra en la FIG. 1, en algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar integrados, cada uno de ellos, con un codificador y un decodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro tipo de hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos diferentes. Si procede, las unidades de MUX-DEMUX pueden ajustarse al protocolo de multiplexado ITU H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0029] La norma ITU-T H.264/MPEG-4 (AVC) fue formulada por el Grupo de Expertos en Codificación de Vídeo de ITU-T (VCEG), junto al Grupo de Expertos en Imágenes en Movimiento de ISO/IEC (MPEG), como el producto de una asociación colectiva conocida como el Equipo de Vídeo Conjunto (JVT). En algunos aspectos, las técnicas descritas en esta divulgación pueden ser aplicadas a dispositivos que se ajustan en general a la norma H.264. La norma H.264 se describe en la Recomendación ITU-T H.264, Codificación de Vídeo Avanzada, para los servicios audiovisuales genéricos, por el Grupo de Estudio de la ITU-T, y con fecha de marzo de 2005, y que se puede denominar en el presente documento norma H.264 o memoria descriptiva H.264, o la norma o memoria descriptiva H.264/AVC. El Equipo de Vídeo Conjunto (JVT) continúa funcionando en extensiones para la norma H.264/MPEG-4 AVC.

[0030] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse cada uno como cualquiera de una variedad de circuitería de codificadores adecuada, tal como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas programables in

situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de estos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio, legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como 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.

[0031] El JCT-VC ha desarrollado la norma HEVC y está funcionando en extensiones escalables y de múltiples vistas para la norma HEVC. Los esfuerzos de normalización de la HEVC se basan en un modelo en evolución de un dispositivo de codificación de vídeo denominado modelo de prueba de la HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a dispositivos existentes de acuerdo con, por ejemplo, la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación de intra-predicción, el HM puede proporcionar hasta treinta y tres modos de codificación de intra-predicción.

[0032] En general, el modelo de funcionamiento del HM describe que una trama o imagen de vídeo puede dividirse en una secuencia de bloques de árbol o unidades de codificación de máximo tamaño (LCU), que incluyen muestras tanto de luma como de croma. Los datos sintácticos dentro de un flujo de bits (bitstream) pueden definir un tamaño para la LCU, que es la máxima unidad de codificación en lo que respecta al número de píxeles. Un fragmento incluye un cierto número de bloques de árbol consecutivos en orden de codificación. Una trama o imagen de vídeo puede dividirse en uno o más fragmentos. Cada bloque de árbol puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuádruple. En general, una estructura de datos de árbol cuádruple incluye un nodo por CU, con un nodo raíz correspondiente al bloque arbolado. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hoja, cada uno de los cuales corresponde a una de las sub-CU.

[0033] Cada nodo de la estructura de datos de árbol cuádruple puede proporcionar datos sintácticos para la CU correspondiente. Por ejemplo, un nodo en el árbol cuádruple puede incluir un indicador (flag) de división, que indica si la CU correspondiente al nodo está dividida o no en varias sub-CU. Los elementos sintácticos para una CU pueden definirse de manera recursiva y pueden depender de si la CU está dividida o no en varias sub-CU. Si una CU no está dividida adicionalmente, se denomina CU hoja. En esta divulgación, cuatro sub-CU de una CU hoja también se denominarán CU hoja aunque no haya una división explícita de la CU hoja original. Por ejemplo, si una CU con un tamaño de 16x16 no se divide adicionalmente, las cuatro sub-CU de tamaño 8x8 también se denominarán CU hoja aunque la CU de tamaño 16x16 no se haya dividido nunca.

[0034] Una CU tiene un propósito similar a un macrobloque de la norma H.264, excepto que una CU no tiene una distinción de tamaño. Por ejemplo, un bloque arbolado puede dividirse en cuatro nodos secundarios (también denominados sub-CU) y cada nodo secundario puede a su vez ser un nodo principal y dividirse en otros cuatro nodos secundarios. Un nodo secundario final, no dividido, denominado un nodo hoja del árbol cuádruple, comprende un nodo de codificación, también denominado CU hoja. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces que puede dividirse un bloque arbolado, denominado profundidad de CU máxima, y también pueden definir un tamaño mínimo de los nodos de codificación. Por consiguiente, un flujo de bits también puede definir una unidad de codificación mínima (SCU). Esta divulgación utiliza el término "bloque" para referirse a cualquiera entre una CU, PU o TU, en el contexto de HEVC, o a estructuras de datos similares en el contexto de otras normas (por ejemplo, macrobloques y subbloques de los mismos en la norma H.264/AVC).

[0035] Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformación (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe ser de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque de árbol, con un máximo de 64x64 píxeles o más. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Los modos de división pueden diferir en función de si la CU está codificada en modo de salto o directa, codificada en modo de intra-predicción o codificada en modo de inter-predicción. Las PU pueden dividirse para no tener forma cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuádruple. Una TU puede tener forma cuadrada o no cuadrada (por ejemplo, rectangular).

[0036] La norma HEVC soporta unas transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU típicamente se basa en el tamaño de las PU de una CU dada definida para una LCU dividida, aunque puede que no siempre sea así. Las TU presentan típicamente el mismo tamaño o un tamaño más pequeño que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas mediante una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformación (TU). Los valores de diferencias de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformación, que pueden cuantificarse.

[0037] Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa una zona espacial correspondiente a la totalidad, o una parte, de la CU correspondiente, y puede incluir datos para recuperar

una muestra de referencia para la PU. Además, una PU incluye datos relacionados con la predicción. Por ejemplo, cuando la PU está codificada en modo de intra, pueden incluirse datos para la PU en un árbol cuádruple residual (RQT), que pueden incluir datos que describen un modo de intra-predicción para una TU correspondiente a la PU. Como otro ejemplo, cuando la PU está codificada en el modo de inter, la PU puede incluir datos que definen uno o

5 más vectores de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, la Lista 0, la Lista 1 o la Lista C) para el vector de movimiento.

10 **[0038]** Una CU hoja que tiene una o más PU también puede incluir una o más unidades de transformación (TU). Las unidades de transformación pueden especificarse usando un RQT (también denominado estructura de árbol cuádruple de TU), como se ha expuesto anteriormente. Por ejemplo, un indicador de división puede indicar si una CU hoja está dividida en cuatro unidades de transformación. A continuación, cada unidad de transformación puede

15 dividirse adicionalmente en más sub-TU. Cuando una TU no se separa adicionalmente, puede denominarse TU hoja. En general, en lo que respecta a la intra-codificación, todas las TU hoja que pertenecen a una CU hoja comparten el mismo modo de intra-predicción. Es decir, el mismo modo de intra-predicción se aplica en general para calcular valores predichos para todas las TU de una CU hoja. En lo que respecta a la intra-codificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando el modo de intra-predicción, como una diferencia

20 entre la parte de la CU correspondiente a la TU y el bloque original. Una TU no está necesariamente limitada al tamaño de una PU. De este modo, las TU pueden ser mayores o menores que una PU. En lo que respecta a la intra-codificación, una PU puede estar co-ubicada con una TU hoja correspondiente para la misma CU. En algunos ejemplos, el tamaño máximo de una TU hoja puede corresponder al tamaño de la CU hoja correspondiente.

25 **[0039]** Además, las TU de las CU hoja también pueden asociarse a las respectivas estructuras de datos de árbol cuádruple, denominadas árboles cuádruples residuales (RQT). Es decir, una CU hoja puede incluir un árbol cuádruple que indica cómo la CU hoja está dividida en varias TU. El nodo raíz de un árbol cuádruple de TU corresponde en general a una CU hoja, mientras que el nodo raíz de un árbol cuádruple de CU corresponde en general a un bloque arbolado (o LCU). Las TU del RQT que no están divididas se denominan TU hoja. En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a

30 no ser que se indique lo contrario.

[0040] Una secuencia de vídeo incluye típicamente una serie de tramas o imágenes de vídeo. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir

35 datos sintácticos en una cabecera del GOP, en una cabecera de una o más de las imágenes o en otras ubicaciones, que describen un cierto número de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen un modo de codificación para el fragmento respectivo. Un codificador de vídeo 20 actúa típicamente sobre bloques de vídeo dentro de fragmentos de vídeo individuales con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los bloques de vídeo pueden presentar tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una

40 norma de codificación especificada.

[0041] Como un ejemplo, el HM soporta la predicción en diversos tamaños de PU. Suponiendo que el tamaño de una CU particular sea $2N \times 2N$, el HM soporta la intra-predicción en tamaños de PU de $2N \times 2N$ o $N \times N$ y la inter-predicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también soporta la división

45 asimétrica para la inter-predicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida en el 25 % y el 75 %. La parte de la CU correspondiente a la división del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así, por ejemplo, " $2N \times nU$ " se refiere a una CU de tamaño $2N \times 2N$ que está dividida

50 horizontalmente con una PU de tamaño $2N \times 0,5N$ encima y una PU de tamaño $2N \times 1,5N$ debajo.

[0042] En esta divulgación, " $N \times N$ " y "N por N" pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de tamaño 16×16 tendrá 16 píxeles en la dirección vertical

55 ($y = 16$) y 16 píxeles en la dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ presenta, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles en un bloque pueden estar ordenados en filas y columnas. Además, no es necesario que los bloques presenten necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N.

60

[0043] Tras la codificación intra-predictiva o inter-predictiva usando las PU de una CU, el codificador de vídeo 20 puede calcular los datos residuales para las TU de la CU. Las PU pueden comprender datos sintácticos que describen un procedimiento o modo de generación de datos de píxeles predictivos en el dominio espacial (también denominado el dominio de píxeles) y las TU pueden comprender coeficientes en el dominio de transformación, tras

65 la aplicación de una transformación, por ejemplo, una transformación de coseno discreta (DCT), una transformación entera, una transformación de ondículas o una transformación conceptualmente similar, a los datos de vídeo

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

5 **[0044]** Tras cualquier transformación para generar coeficientes de transformación, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformación. La cuantificación se refiere, en general, a un proceso en el que los coeficientes de transformación se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando compresión adicional. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes. Por ejemplo, un valor de n bits puede redondearse hacia abajo hasta un valor de m bits durante la cuantificación, donde n es mayor que m .

15 **[0045]** Después de la cuantificación, el codificador de vídeo puede escanear los coeficientes de transformación, produciendo un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformación cuantificados. El escaneo puede estar diseñado para colocar los coeficientes de energía más alta (y por lo tanto de menor frecuencia) en la parte frontal de la matriz y para colocar los coeficientes de energía más bajos (y por lo tanto de mayor frecuencia) en la parte posterior de la matriz. En algunos ejemplos, el codificador de vídeo 20 puede usar un orden de escaneo predefinido para escanear los coeficientes de transformación cuantificados, para producir un vector en serie que pueda ser codificado por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar un escaneo adaptativo. Después de escanear los coeficientes de transformación cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa de acuerdo con el contexto (CAVLC), la codificación aritmética binaria adaptativa de acuerdo con el contexto (CABAC), la codificación aritmética binaria adaptativa de acuerdo con el contexto basada en la sintaxis (SBAC), la codificación por entropía por división de intervalos de probabilidad (PIPE) u otros procedimientos de codificación por entropía. El codificador de vídeo 20 también puede realizar la codificación por entropía de los elementos sintácticos asociados a los datos de vídeo codificados para su uso por el descodificador de vídeo 30 en la descodificación de los datos de vídeo.

30 **[0046]** Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto dentro de un modelo contextual a un símbolo que se va a transmitir. El contexto puede referirse, por ejemplo, a si los valores contiguos del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo que se va a transmitir. Las palabras de código en la VLC pueden construirse de forma que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta manera, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo que se va a transmitir. La determinación de la probabilidad puede basarse en un contexto asignado al símbolo.

40 **[0047]** De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 y/o el descodificador de vídeo 30 pueden estar configurados para implementar técnicas de una o las dos categorías analizadas anteriormente. Es decir, el codificador de vídeo 20 y/o el descodificador de vídeo 30 pueden configurarse para realizar técnicas de la primera categoría (que se refiere a la restricción de la predicción entre capas) y de la segunda categoría (que se refiere a la indicación).

45 **[0048]** Varios ejemplos de técnicas de la primera categoría se explican a continuación. Estas técnicas se pueden usar solas o en cualquier combinación, entre ellas y/o con técnicas de la segunda categoría.

50 **[0049]** En un primer ejemplo de las técnicas de la primera categoría, los límites de los elementos de una capa de referencia y capas de mejora pueden estar alineados. Sin pérdida de generalidad, la FIG. 9 (descrita a continuación) muestra que se usan 4 elementos para capas de referencia y mejora. En este caso, el codificador de vídeo 20 y/o el descodificador de vídeo 30 pueden realizar la codificación con 4 núcleos de procesador, cada uno de los cuales está dedicado a uno de los elementos respectivos. Para garantizar el procesamiento paralelo de los 4 elementos, no se permite que el muestreo ascendente del filtrado entre capas cruce los límites de los elementos en la capa de referencia. Para un elemento de mejora, las muestras colocadas en la capa de referencia se consideran no disponibles si las muestras de la capa de referencia pertenecen a los elementos que no están alineados con este elemento de mejora. Por ejemplo, en la FIG. 9, cuando se realiza un muestreo ascendente o un filtrado entre capas para el elemento E0, las muestras en el elemento B1 se consideran no disponibles. En consecuencia, puede ser necesario un relleno a lo largo de los límites de los elementos de capa de referencia para el muestreo ascendente o el filtrado entre capas. Relleno se refiere a rellenar los valores para los píxeles en el otro lado del límite del elemento, por ejemplo, agregando valores predeterminados a estos píxeles o ampliando los valores del bloque a lo largo del límite del elemento.

65 **[0050]** En otras palabras, la predicción entre capas puede requerir un grado de muestreo ascendente, que típicamente utiliza píxeles de un elemento de capa base para generar valores para píxeles muestreados de forma ascendente para generar un elemento predictivo de una resolución más alta. De acuerdo con este primer ejemplo de la primera categoría, cuando un bloque de la capa de mejora corresponde a un bloque en un límite de elemento en

la capa base, y cuando el muestreo ascendente utilizaría píxeles que cruzan el límite de elemento, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden en cambio tratar los píxeles a lo ancho del límite del elemento como no disponibles, y rellenar valores para estos píxeles.

5 **[0051]** De manera similar, cuando se alinean los límites de los elementos de capa de referencia y las capas de mejora, los filtros de bucle, incluyendo desbloqueo y desplazamiento adaptativo de muestras (SAO), no pueden cruzar los límites de los elementos en capa de referencia para habilitar la codificación / decodificación en paralelo; esto podría cumplirse restringiendo el `loop_filter_across_tiles_enabled_flag` de la capa de referencia a cero.

10 **[0052]** De manera similar, en este primer ejemplo, no se permite que la predicción de sintaxis entre capas cruce los límites de los elementos transversales en la capa de referencia. Como ejemplo, en una solución SHVC basada en intraBL, la fusión colocada de la capa de referencia o el candidato de movimiento AMVP pueden marcarse como no disponibles cuando el bloque colocado está ubicado en el elemento no alineado. Como otro ejemplo más, en soluciones SHVC basadas en índice de referencia (sintaxis de alto nivel solamente), la información de movimiento de la región cercana al límite del elemento (por ejemplo, dentro de un número predefinido de píxeles, CU, PU o similares) puede reiniciarse o recortarse durante el muestreo ascendente / asignación del campo de movimiento.

20 **[0053]** Como una segunda técnica de ejemplo en la primera categoría, se aplica una restricción de codificador de modo que ninguna CU / PU en un elemento de mejora se prediga mediante textura entre capas o se prediga mediante sintaxis entre capas a partir de una CU / PU colocada que puede tener filtrado entre capas o filtrado de capa de referencia a través de los elementos en la imagen de referencia en la misma unidad de acceso. Para indicar esto, se puede usar un indicador de información de usabilidad de vídeo (VUI), como se muestra en la Tabla 1 a continuación, donde el texto *en cursiva* representa adiciones sobre la sintaxis de HEVC:

25 TABLA 1

vui_parameters() {	Descriptor
...	
bitstream_restriction_flag	u(1)
if (bitstream_restriction_flag) {	
tiles_fixed_structure_flag	u(1)
<i>inter_layerpred_not_along_tile_boundary_flag</i>	u(1)
<i>motion_vectors_over_pic_boundaries_flag</i>	u(1)
<i>restricted_ref_pic_lists_flag</i>	u(1)
mins_patial_segmentation_idc	ue(v)
max_bytes_per_pic_denom	ue(v)
max_bits_per_min_cu_denom	ue(v)
log2_max_mv_length_horizontal	ue(v)
log2_max_mv_length_vertical	ue(v)
}	
}	

30 **[0054]** Por lo tanto, la Tabla 1 incluye el elemento sintáctico extra `inter_layerpred_not_along_tile_boundary_flag`. La semántica de este elemento sintáctico se puede definir de la forma siguiente: ***inter_layerpred_not_along_tile_boundary_flag*** igual a 1 indica que los límites de los elementos de las diferentes capas están alineados y no se permite la predicción de texturas entre capas a lo largo de los límites de los elementos para las CU o PU de capa de mejora cuando las CU o PU de capa de referencia colocada pueden tener filtrado entre capas o filtrado de capa de referencia a través de los elementos. El filtrado entre capas también puede incluir SAO entre capas, muestreo entre capas, filtro de bucle entre capas, etc. El filtrado de capa de referencia puede incluir SAO de capa de referencia, filtro de bucle de capa de referencia, etc. La alineación de los límites de los elementos significa que, para dos muestras cualesquiera de imagen de capa de mejora que se encuentren dentro del mismo elemento de mejora, las muestras de capa de referencia colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de referencia y para dos muestras de imagen de capa de referencia que se encuentren dentro del mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de mejora. El elemento sintáctico `inter_layerpred_not_along_tile_boundary_flag` igual a 0 indica que no hay restricciones en las configuraciones de elementos entre capas. Cuando `inter_layerpred_not_along_tile_boundary_flag` no está presente en un flujo de bits, se denominará igual a 0.

45 **[0055]** De forma alternativa, la semántica para este elemento sintáctico se puede definir de la forma siguiente: ***inter_layerpred_not_along_tile_boundary_flag*** igual a 1 indica que los límites de los elementos de las diferentes capas están alineados y no se permite la predicción de texturas entre capas y la predicción de sintaxis entre capas a lo largo de los límites de los elementos, de modo que cualquier elemento de capa de mejora puede descodificarse sin descodificar ningún elemento de capa base no alineado. La alineación de los límites de los

elementos significa que para dos muestras cualesquiera de imágenes de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa de referencia colocadas, si están disponibles, también estarán dentro del mismo elemento de capa de referencia, y para dos muestras cualesquiera de imágenes de capa de referencia que se encuentren dentro del mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de mejora. El elemento sintáctico `inter_layerpred_not_along_tile_boundary_flag` igual a 0 indica que no hay restricciones en las configuraciones de elementos entre capas. Cuando `inter_layerpred_not_along_tile_boundary_flag` no está presente en un flujo de bits, se denominará igual a 0.

[0056] De forma alternativa, el indicador puede indicarse en un mensaje SEI que se aplica a la secuencia de vídeo codificada que contiene el mensaje SEI con semántica similar.

[0057] En otras palabras, cuando el codificador de vídeo 20 o descodificador de vídeo 30 está codificando (por ejemplo, codificando o descodificando) un bloque de capa de mejora en un elemento de capa de mejora de una unidad de acceso, y cuando está habilitado el filtrado entre capas o filtrado de capa de referencia a través de los elementos en una imagen de referencia de la unidad de acceso, el codificador de vídeo 20 o el descodificador de vídeo 30 pueden evitar (por ejemplo, inhabilitar) la codificación del bloque de capa de mejora usando predicción de textura entre capas o predicción de sintaxis entre capas desde un bloque de capa base colocada de la imagen de referencia.

[0058] Como una tercera técnica de ejemplo en la primera categoría, en escalabilidad espacial extendida, la capa de referencia y la capa de mejora puede tener diferentes relaciones de aspecto de imagen, tal como 4:3 para la capa de referencia y 16:9 para la capa de mejora. Es decir, la capa base puede incluir datos de vídeo suficientes para producir un vídeo de relación de aspecto 4:3, mientras que la capa de mejora puede incluir datos de vídeo suficientes para producir un vídeo de relación de aspecto de 16:9. En este caso, puede haber una gran región de capa de mejora que no tiene contraparte en la capa de referencia, de modo que no se puede hacer predicción entre capas para tales regiones (es decir, la región en las imágenes de capa de mejora que se encuentra fuera de la región de relación de aspecto 4:3 de la capa base). Cuando tales regiones (en las imágenes de capa de mejora que se encuentran fuera de la región de relación de aspecto 4:3 de la capa base) y las regiones que pueden predecirse entre capas se codifican en diferentes elementos, se puede obtener un mejor procesamiento paralelo entre capas.

[0059] Para indicar esto en flujo de bits, un nuevo mensaje SEI, denominado mensaje SEI de información de predicción entre capas de elemento en el ejemplo de, con la siguiente sintaxis y la semántica descritas con respecto a la Tabla 2, puede codificarse:

TABLA 2

tile_interlayer_pred_info(payloadSize) {	Descriptor
sei_pic_parameter_set_id	ue(v)
for(i = 0; i <= num_tile_columns_minus1; i++)	
for(j = 0; j <= num_tile_rows_minus1; j++)	
non_interlayer_pred_flag[i][j]	u(1)
}	

[0060] En este ejemplo, **sei_pic_parameter_set_id** especifica el valor de `pps_pic_parameter_set` para el PPS al que hace referencia la imagen asociada con el mensaje SEI de información de predicción entre capas del elemento. El valor de `sei_pic_parameter_set_id` estará dentro de la gama de 0 y 63, inclusive.

[0061] De forma alternativa, `sei_pic_parameter_set_id` puede codificarse mediante longitud fija, por ejemplo, como `u(6)`, en lugar de codificarse como `ue(v)`.

[0062] En este ejemplo, **non_interlayer_pred_flag[i][j]** igual a 1 indica que no se usa ninguna predicción entre capas de textura o sintaxis para el elemento en la i-ésima columna del elemento y la j-ésima fila del elemento. **non_interlayer_pred_flag[i][j]** igual a 0 indica que se puede usar predicción entre capas de textura o sintaxis para el elemento en la i-ésima columna del elemento y la j-ésima fila del elemento.

[0063] Como cuarta técnica de ejemplo en la primera categoría, en la 13^a reunión de JCTVC en Incheón, Corea, el indicador de alineación de límites (boundaries) de los elementos se ha adoptado como en JCTVC-M0464 para extensiones de HEVC. La siguiente técnica, más específicamente, está relacionada con la alineación de límites de los elementos para escalabilidad espacial extendida y escenarios de uso de escalabilidad de relación de aspecto.

[0064] En este ejemplo, la semántica del indicador de alineación de límites de los elementos se modifica de la forma siguiente, donde el texto entre corchetes precedido por "eliminado" indica las supresiones de la definición anterior: `tile_boundaries_aligned_flag` igual a 1 indica que [eliminado: "todos los límites de los elementos de la imagen de capa base correspondiente tienen límites de los elementos correspondientes en la capa de mejora dada y no existen límites de los elementos adicionales en la capa de mejora. Eso significa que, "] para dos muestras cualesquiera de imagen de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa de referencia colocadas, si están disponibles, también estarán dentro del mismo elemento de capa de referencia y para dos muestras cualesquiera de imagen de capa de referencia que se encuentren dentro el mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también estarán dentro del mismo elemento de capa de mejora. `tile_boundaries_aligned_flag` igual a 0 indica que no hay restricción en la configuración del elemento entre la capa de mejora correspondiente y la capa de referencia dada.

[0065] También, puede observarse que, para los casos escalabilidad espacial extendida, puede haber límites de los elementos adicionales en la capa de mejora en comparación con la capa de referencia y también, de forma alternativa, puede haber límites de los elementos adicionales en la capa de referencia en comparación con la capa de mejora.

[0066] Como una quinta técnica de ejemplo en la primera categoría, de forma alternativa, un indicador de alineación de límite de elemento puede estar normativamente restringido a cero para los casos de escalabilidad extendida o uso de escalabilidad de relación extendida. En este ejemplo, la siguiente restricción se agrega a `tile_boundaries_aligned_flag`: `tile_boundaries_aligned_flag` debe ser igual a cero cuando `extended_spatial_scalability_idc` es mayor que cero. De forma alternativa, `tile_boundaries_aligned_flag` será igual a cero cuando al menos uno de los parámetros geométricos para el proceso de remuestreo (resampling) tenga un valor distinto de cero. Los parámetros geométricos incluyen `scaled_ref_layer_left_offset`, `scaled_ref_layer_top_offset`, `scaled_ref_layer_right_offset`, `scaled_ref_layer_bottom_offset`.

[0067] La segunda categoría de técnicas se refiere a la indicación para el procesamiento paralelo. SHVC emplea el marco de descodificación escalable de múltiples bucles, donde todas las imágenes de capa de referencia de las que depende la imagen de capa actual (aunque cualquier tipo de predicción entre capas) deben descodificarse para comenzar a descodificar la imagen de capa actual. Sin embargo, los descodificadores pueden procesar diferentes capas en paralelo. Puede ser beneficioso si un hilo descodificador para una capa de mejora puede comenzar a descodificar tan pronto como se completa la descodificación de la región parcial de una imagen de capa de referencia. Esta descodificación parcial de una imagen de capa de referencia puede corresponder a uno o más elementos.

[0068] También se puede indicar que los elementos más pequeñas pueden aumentar el paralelismo, pero al mismo tiempo, reducir la eficiencia de codificación, y viceversa. Por lo tanto, para el caso de escalabilidad espacial, las imágenes de capa de referencia y mejora podrían usar estructuras de elemento diferentes. En este caso, puede ser deseable que el descodificador tenga información que define el número máximo de elementos en una imagen de capa de referencia que necesita descodificarse antes de comenzar a descodificar un elemento de capa de mejora, por ejemplo, para permitir (to enable) una mayor reducción de retardo de códec total.

[0069] En un primer ejemplo la técnica de la segunda categoría, el codificador de vídeo 20 puede indicar, y el descodificador de vídeo 30 puede recibir, un valor tal usando un nuevo elemento sintáctico asociado con cada elemento en una capa de mejora, por ejemplo, el nombre `max_ref_tile_dec_idc_minus1[i]`. La semántica de este elemento sintáctico puede ser la siguiente: `max_ref_tile_dec_idc_minus1[i]` más 1 indica el número máximo de elementos correspondientes en una imagen de capa de referencia de la imagen de capa actual que debe descodificarse antes de que se pueda descodificar el elemento actual.

[0070] El descodificador de vídeo 30 puede obtener los elementos correspondientes como elementos consecutivos basándose en el orden de escaneado de elementos de la imagen de capa de referencia. El elemento sintáctico se puede indicar en el conjunto de parámetros de vídeo (VPS), conjunto de parámetros de secuencia (SPS), conjunto de parámetros de imagen (PPS), cabecera de fragmento, información de usabilidad de vídeo (VUI) o como un mensaje de información de mejora suplementaria (SEI). u otras estructuras sintácticas, preferentemente condicionadas basándose en `tile_enable_flag`, y no indicadas para una imagen de capa con `nuh_layer_id` igual a 0.

[0071] En una segunda técnica de ejemplo de la segunda categoría (que se expande en la primera técnica de ejemplo de la segunda categoría), el codificador de vídeo 20 puede indicar y el descodificador de vídeo 30 puede recibir, un valor usando un nuevo elemento sintáctico asociado con cada elemento en una capa de mejora y para cada imagen de capa de referencia de la que depende la imagen de capa de mejora actual, por ejemplo, denominada `max_ref_tile_dec_idc_minus1[i][j]`. La semántica para este elemento sintáctico puede ser la siguiente: `max_ref_tile_dec_idc_minus1[i][j]` más 1 indica el número máximo de elementos correspondientes en una ^{j-ésima} imagen de capa de referencia de la imagen de capa actual que debe descodificarse antes de que el ^{i-ésimo} elemento actual se pueda descodificar.

[0072] El descodificador de vídeo 30 puede obtener los elementos correspondientes como elementos consecutivos

basándose en el orden de escaneado de elementos de la imagen de capa de referencia. El elemento sintáctico se puede indicar en el VPS, SPS, PPS, cabecera de división, VUI o un mensaje SEI u otras estructuras sintácticas, preferentemente condicionadas basándose en `tile_enable_flag`, y no indicadas para una imagen de capa con `nuh_layer_id` igual a 0.

[0073] En una tercera técnica de ejemplo de la segunda categoría (que puede extenderse en la primera técnica de ejemplo en la segunda categoría), los elementos correspondientes de un elemento capa de mejora, también pueden indicarse mediante la indicación explícita de los ID de elemento de los elementos correspondientes en el flujo de bits, por ejemplo, en el VPS, SPS, PPS, cabecera de fragmento, VUI o un mensaje SEI u otras estructuras sintácticas.

[0074] En una cuarta técnica de ejemplo de la segunda categoría (que puede extenderse en la primera y/o segunda técnicas de ejemplo en la segunda categoría), el proceso de escaneado de elementos en la capa de referencia puede ser indicado para una capa de mejora para permitir un proceso de escaneado de elementos arbitrario.

[0075] Una quinta técnica de ejemplo de la segunda categoría se refiere al filtrado de entre capas través de un límite de elemento. De forma similar al indicador `loop_filter_across_tiles_enabled_flag` definido en el borrador de HEVC, para habilitar el procesamiento paralelo mejorado entre los elementos de diferentes capas, se propone definir un indicador, por ejemplo, llamada `interlayer_filter_across_tiles_enabled_flag`, con la siguiente semántica: **`interlayer_filter_across_tiles_enabled_flag`** igual a 1 especifica que las operaciones de filtrado entre capas se realizan a lo ancho de los límites de los elementos de capa de referencia. `interlayer_filter_across_tiles_enabled_flag` igual a 0 especifica que las operaciones de filtrado entre capas no se realizan a lo ancho de los límites de los elementos. Las operaciones de filtrado entre capas pueden incluir SAO entre capas, ALF entre capas y/o muestreo ascendente. El indicador se puede indicar en el VPS, SPS, PPS, cabecera de fragmento u otras estructuras sintácticas.

[0076] Una sexta técnica de ejemplo de la segunda categoría se refiere a la predicción entre capas través de un límite de elemento. En esta sexta técnica de ejemplo, se propone definir un indicador, por ejemplo, llamado `interlayer_pred_across_tiles_enabled_flag`, con la siguiente semántica: **`interlayer_pred_across_tiles_enabled_flag`** igual a 1 especifica que las operaciones de predicción entre capas se realizan a lo ancho de los límites de los elementos de capa de referencia. `interlayer_pred_across_tiles_enabled_flag` igual a 0 especifica que las operaciones de filtrado entre capas no se realizan a lo ancho de los límites de los elementos. La predicción entre capas puede incluir predicción de textura, movimiento, sintaxis o residuos. El indicador se puede indicar en el VPS, SPS, PPS, cabecera de fragmento u otras estructuras sintácticas.

[0077] Esta divulgación también describe técnicas relacionadas con el muestreo ascendente basado en elementos. El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden estar configurados para implementar técnicas de muestreo ascendente basadas en elementos de esta divulgación. En la práctica, tal vez no sea preferible que el muestreo ascendente no se permita a lo largo y/o a lo ancho de los límites de los elementos. Es decir, puede ser preferible permitir el muestreo ascendente a lo largo y/o a lo ancho de los límites de los elementos, de acuerdo con las técnicas de esta divulgación.

[0078] Como un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar un elemento sintáctico (por ejemplo, un indicador en un SPS, PPS, o VPS, para una o más capas de mejora) que indica si el muestreo ascendente basado en elementos está habilitado (por ejemplo, para una capa de mejora correspondiente). Por ejemplo, la sintaxis que se muestra en la Tabla 3 a continuación puede indicarse en un SPS, donde el elemento sintáctico *en cursiva* representa una adición con respecto a la sintaxis convencional para el SPS:

TABLA 3

<code>if(sps_extension_flag) {</code>	
...	
<i><code>resampling_filter_across_tiles_enabled_flag</code></i>	<code>u(1)</code>
...	

[0079] La semántica para este elemento sintáctico se puede definir de la forma siguiente: **`resampling_filter_across_tiles_enabled_flag`** igual a 0 especifica que el muestreo ascendente basado en imágenes está habilitado. `resampling_filter_across_tiles_enabled_flag` igual a 1 especifica que el muestreo ascendente basado en elementos está habilitado.

[0080] Cuando el muestreo ascendente basado en elementos está habilitado, el codificador de vídeo 20 y el

descodificador de vídeo 30 pueden tratar los límites de los elementos de la misma manera que los límites de la imagen en el proceso de muestreo ascendente. Es decir, cuando se necesita una muestra de píxeles fuera del elemento actual (que puede estar dentro o fuera de la imagen actual) para el muestreo ascendente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden generar un valor para el píxel con relleno de píxeles a partir de esos píxeles en el interior el elemento actual. El proceso de relleno puede ser el mismo que el definido para los píxeles fuera de la imagen actual, por ejemplo, mediante HEVC.

[0081] Principios similares también pueden aplicarse a filtrado entre capas. Las operaciones de filtrado entre capas pueden incluir ALF entre capas y/o filtro de suavizado para SNR, y similares. La Tabla 4 siguiente representa un ejemplo de datos sintácticos indicativos de si las operaciones de filtrado entre capas están habilitadas, donde el texto en cursiva representa adiciones a la versión anterior de la norma propuesta.

TABLA 4

if(sps_extension_flag) {	
...	
<i>inter_layer_filter_across_tiles_enabled_flag</i>	u(1)
...	

[0082] La semántica para este elemento sintáctico se puede definir de la forma siguiente: ***inter_layer_filter_across_tiles_enabled_flag*** igual a 0 especifica que el filtrado entre capas basado en imágenes está habilitado. ***inter_layer_filter_across_tiles_enabled_flag*** igual a 1 especifica que el filtrado entre capas basado en elementos está habilitado. Cuando no está presente (por ejemplo, en un SPS), el descodificador de vídeo 30 puede deducir que el valor de ***inter_layer_filter_across_tiles_enabled_flag*** es igual a 0. De manera similar al proceso de muestreo ascendente basado en elementos, cuando el filtrado entre capas basado en elementos está habilitado, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden tratar los límites de los elementos de la misma manera que los límites de las imágenes en el proceso de filtrado entre capas.

[0083] Esta divulgación también describe técnicas para muestreo ascendente usando píxeles reconstruidos antes de SAO y filtros de desbloqueo alrededor de los límites de los elementos y/o fragmentos. En un PPS, ***loop_filter_across_tiles_enabled_flag*** igual a 1 puede especificar que las operaciones de filtrado en bucle se pueden realizar a lo ancho de los límites de los elementos en las imágenes que se refieren al PPS. ***loop_filter_across_tiles_enabled_flag*** igual a 0 puede especificar que las operaciones de filtrado en bucle no se realizan a lo ancho de los límites de los elementos en las imágenes que se refieren al PPS. Las operaciones de filtrado en bucle pueden incluir operaciones de filtro de desbloqueo y filtro de desplazamiento adaptativo de muestra. Cuando no está presente, se deduce que el valor de ***loop_filter_across_tiles_enabled_flag*** es igual a 1.

[0084] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar un elemento sintáctico (por ejemplo, un indicador en un SPS o VPS, para cada capa de mejora) que indique si el muestreo ascendente basado en elementos se realiza utilizando los píxeles reconstruidos antes (o después) de realizarse un proceso de filtrado de bucle de capa de referencia. La Tabla 5 representa un ejemplo de datos sintácticos que se pueden indicar en un SPS, indicativo de si el muestreo ascendente basado en elementos se realiza utilizando los píxeles reconstruidos antes del proceso de filtrado del bucle de la capa de referencia, donde el texto *en cursiva* representa las adiciones propuestas por esta divulgación a la versión anterior de la norma propuesta.

TABLA 5

if(sps_extension_flag) {	
...	
if(loop_filter_across_tiles_enabled_flag)	
<i>resampling_filter_before_loop_filter_across_tiles_enabled_flag</i>	u(1)
...	

[0085] La semántica para este elemento sintáctico se puede definir de la forma siguiente: ***resampling_filter_before_loop_filter_across_tiles_enabled_flag*** igual a 0 especifica que el muestreo ascendente basado en elementos utiliza los píxeles reconstruidos antes del proceso de filtrado del bucle de la capa de referencia. ***resampling_filter_across_tiles_enabled_flag*** igual a 1 especifica que el muestreo ascendente basado en elementos usa los píxeles reconstruidos después del filtrado del bucle de la capa de referencia. Cuando no está presente, el descodificador de vídeo 30 puede deducir que el valor de ***resampling_filter_across_tiles_enabled_flag***

es igual a 1. Principios similares se pueden aplicar al filtrado entre capas. Del mismo modo, se pueden aplicar principios similares a los límites de fragmento, además o en lugar de los límites de los elementos. El término "unidad de sub-imagen" puede referirse a un fragmento, elemento, frente de onda, fila de CTU, u otra unidad de ese tipo que sea más pequeña que una imagen completa e incluya una pluralidad de CTU. Por lo tanto, las técnicas anteriores pueden describirse como técnicas para codificar datos que indican si el muestreo ascendente basado en unidades de sub-imagen se realiza utilizando píxeles reconstruidos antes de que se realice un proceso de filtrado de bucle de capa de referencia.

[0086] Esta divulgación también describe técnicas relacionadas con el muestreo ascendente basado en el procesamiento paralelo de frente de onda (WPP). En la práctica, para aplicaciones de bajo retardo que usan WPP, no se prefiere que el muestreo ascendente (o el filtrado suave para el caso de SNR) no se permita a lo largo de los límites verticales de la CTU. Es decir, puede ser deseable permitir el muestreo ascendente (o el filtrado uniforme) a lo largo de los límites verticales de la CTU. Como un ejemplo de implementación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar relleno para evitar el uso de datos de las CTU de abajo y de la derecha cuando se realiza el filtrado entre capas. Esta implementación solo requiere que la CTU de capa de referencia se codifique antes de la CTU de capa de mejora.

[0087] De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar configurados para codificar un elemento sintáctico (por ejemplo, un indicador en un SPS o VPS, para cada capa de mejora) que indica si el filtrado entre capas basado en WPP está habilitado. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos sintácticos en un SPS de acuerdo con la Tabla 6 siguiente, donde el texto *en cursiva* representa las adiciones propuestas por esta divulgación.

TABLA 6

if(sps_extension_flag) {	
...	
if(entropy_coding_sync_enabled_flag)	
<i>resampling_filter_across_wpps_enabled_flag</i>	u(1)
...	

[0088] La semántica para este elemento sintáctico se puede definir de la forma siguiente: ***resampling_filter_across_wpps_enabled_flag*** igual a 0 especifica que el muestreo ascendente basado en imágenes está habilitado. ***resampling_filter_across_wpps_enabled_flag*** igual a 1 especifica que el muestreo ascendente basado en filas de CTU está habilitado. Cuando no está presente, el decodificador de vídeo 30 puede deducir que el valor de ***resampling_filter_across_wpps_enabled_flag*** es igual a 0.

[0089] Cuando el muestreo ascendente basado en filas de CTU está habilitado, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden tratar los límites de fila de CTU la misma manera que los límites de imagen en un proceso de muestreo ascendente. Es decir, cuando se necesita una muestra de píxeles fuera de la fila de CTU actual (que puede estar dentro o fuera de la imagen actual) para el muestreo ascendente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden generar un valor para el píxel con relleno de píxeles de esos píxeles dentro de la fila de CTU actual. El proceso de relleno puede ser el mismo que el definido para los píxeles fuera de la imagen actual, por ejemplo, de acuerdo con HEVC.

[0090] Además, el codificador de vídeo 20 puede enviar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en tramas y datos sintácticos basados en GOP, al decodificador de vídeo 30, por ejemplo, en una cabecera de trama, una cabecera de bloque, una cabecera de fragmento o una cabecera de GOP. Los datos sintácticos de GOP pueden describir un cierto número de tramas en el respectivo GOP, y los datos sintácticos de trama pueden indicar un modo de codificación/predicción utilizada para codificar la trama correspondiente.

[0091] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse, cada uno, como cualquiera entre una amplia variedad de circuitería de codificadores o decodificadores adecuada, según corresponda, tal como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables en el terreno (FPGA), circuitería de lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 puede estar incluido en uno o más codificadores o decodificadores, cada uno de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) de vídeo combinado. Un dispositivo que incluye el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

[0092] La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20, que puede implementar las técnicas para procesamiento de datos de vídeo en paralelo. El codificador de vídeo 20 puede realizar la codificación intra y la codificación inter de bloques de vídeo dentro de fragmentos de vídeo. La codificación intra se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama o imagen de vídeo dada. La codificación inter se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El modo de intra (modo I) puede referirse a cualquiera de varios modos de codificación de base espacial. Los modos de inter, tales como la predicción unidireccional (modo P) o la predicción bidireccional (modo B), pueden referirse a cualquiera de varios modos de codificación de base temporal.

[0093] Como se muestra en la FIG. 2, el codificador de vídeo 20 recibe un bloque de vídeo actual dentro de una trama de vídeo a codificar. En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una unidad de selección de modo 40, una memoria de imágenes de referencia 64, un sumador 50, una unidad de procesamiento de transformación 52, una unidad de cuantificación 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modo 40 incluye una unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de intra-predicción 46 y una unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye además la unidad de cuantificación inversa 58, la unidad de transformación inversa 60 y el sumador 62. También puede incluirse un filtro de desbloqueo (no se muestra en la FIG. 2) para filtrar límites de bloque, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará típicamente la salida del sumador 62. También pueden usarse filtros adicionales (en bucle o pos-bucle), además del filtro de desbloqueo. Dichos filtros no se muestran por razones de brevedad pero, si se desea, pueden filtrar la salida del sumador 50 (tal como un filtro en bucle).

[0094] Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un fragmento de vídeo que va a codificarse. La trama o el fragmento puede estar dividido en múltiples bloques de vídeo. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 llevan a cabo la codificación inter-predictiva del bloque de vídeo recibido con respecto a uno o más bloques en una o más tramas de referencia, para proporcionar la predicción temporal. La unidad de intra-predicción 46, de forma alternativa, puede llevar a cabo la codificación intra-predictiva del bloque de vídeo recibido, con respecto a uno o más bloques contiguos de la misma trama o fragmento que el bloque que va a codificarse, para proporcionar la predicción espacial. El codificador de vídeo 20 puede llevar a cabo múltiples pases de codificación, por ejemplo, para seleccionar un modo de codificación adecuada para cada bloque de datos de vídeo.

[0095] Además, la unidad de división 48 puede dividir bloques de datos de vídeo en sub-bloques, basándose en la evaluación de los anteriores esquemas de división en los pases de codificación anteriores. Por ejemplo, la unidad de división 48 puede dividir inicialmente una trama o un fragmento en varias LCU, y dividir cada una de las LCU en varias sub-CU, basándose en un análisis de velocidad-distorsión (por ejemplo, optimización de velocidad-distorsión). La unidad de selección de modo 40 puede producir además una estructura de datos de árbol cuádruple que indica la división de una LCU en varias sub-CU. Las CU de nodos hoja del árbol cuádruple pueden incluir una o más PU y una o más TU.

[0096] La unidad de selección de modo 40 puede seleccionar uno de los modos de codificación, intra o inter, por ejemplo, basándose en los resultados de errores, y proporciona el bloque intra-codificado o inter-codificado resultante al sumador 50 para generar datos de bloque residuales y al sumador 62 para reconstruir el bloque codificado para su uso como una trama de referencia. La unidad de selección de modo 40 proporciona además elementos sintácticos, tales como vectores de movimiento, indicadores de modo de intra, información de división y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

[0097] La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento de los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen actual de vídeo, con respecto a un bloque predictivo dentro de una trama de referencia (u otra unidad codificada), con respecto al bloque actual que está siendo codificado dentro de la trama actual (u otra unidad codificada). Un bloque predictivo es un bloque que se revela como estrechamente coincidente con el bloque a codificar en lo que respecta a la diferencia de píxeles, que puede determinarse mediante la suma de una diferencia absoluta (SAD), una suma de diferencia de cuadrados (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de imágenes de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones de píxel fraccionarias de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento en relación con las posiciones de píxeles completas y las posiciones de píxeles fraccionarias, y emitir un vector de movimiento con una precisión de píxel fraccionaria.

[0098] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de

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

[0099] La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar capturar o generar el bloque predictivo basándose en el vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden integrarse funcionalmente, en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que está siendo codificado, generando valores de diferencia de píxel, como se expone posteriormente. En general, la unidad de estimación de movimiento 42 lleva a cabo la estimación de movimiento con respecto a los componentes de luma, y la unidad de compensación de movimiento 44 utiliza los vectores de movimiento calculados basándose en los componentes de luma, tanto para los componentes de croma como para los componentes de luma. La unidad de selección de modo 40 también puede generar elementos sintácticos asociados a los bloques de vídeo y al fragmento de vídeo para su uso por parte del descodificador de vídeo 30 a la hora de descodificar los bloques de vídeo del fragmento de vídeo.

[0100] La unidad de intra-predicción 46 puede intra-predecir un bloque actual, de forma alternativa a la inter-predicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de intra-predicción 46 puede determinar un modo de intra-predicción que se va a utilizar para codificar un bloque actual. En algunos ejemplos, la unidad de intra-predicción 46 puede codificar un bloque actual usando varios modos de intra-predicción, por ejemplo, durante diferentes pases de codificación, y la unidad de intra-predicción 46 (o la unidad de selección de modo 40, en algunos ejemplos) puede seleccionar un modo adecuado de intra-predicción para utilizar a partir de los modos probados.

[0101] Por ejemplo, la unidad de intra-predicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intra-predicción probados, y seleccionar el modo de intra-predicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina en general una magnitud de distorsión (o errores) entre un bloque codificado y un bloque original, no codificado, que se codificó para generar el bloque codificado, así como una velocidad de bits (es decir, un cierto número de bits) usada para generar el bloque codificado. La unidad de intra-predicción 46 puede calcular razones a partir de las distorsiones y velocidades para los diversos bloques codificados, para determinar qué modo de intra-predicción presenta el mejor valor de velocidad-distorsión para el bloque.

[0102] Después de seleccionar un modo de intra-predicción para un bloque, la unidad de intra-predicción 46 puede proporcionar información indicativa del modo de intra-predicción seleccionado para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información indicando el modo de intra-predicción seleccionado. El codificador de vídeo 20 puede incluir datos de configuración en el flujo de bits transmitido, que pueden incluir una pluralidad de tablas de índices de modos de intra-predicción y una pluralidad de tablas de índices de modos de intra-predicción modificadas (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para varios bloques e indicaciones del modo de intra-predicción más probable, una tabla de índices de modos de intra-predicción y una tabla de índices de modos de intra-predicción modificados a utilizar para cada uno de los contextos.

[0103] El codificador de vídeo 20 forma un bloque de vídeo residual sustrayendo los datos de predicción de la unidad de selección de modo 40 del bloque de vídeo original que está codificándose. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformaciones 52 aplica una transformación, tal como una transformación discreta del coseno (DCT) o una transformación conceptualmente similar al bloque residual, generando un bloque de vídeo que comprende valores residuales de coeficientes de transformación. La unidad de procesamiento de transformaciones 52 puede llevar a cabo otras transformaciones que son conceptualmente similares a la DCT. También podrían usarse transformaciones de ondículas, transformaciones de enteros, transformaciones de sub-bandas u otros tipos de transformaciones. En cualquier caso, la unidad de procesamiento de transformaciones 52 aplica la transformación al bloque residual, generando un bloque de coeficientes de transformación residuales. La transformación puede convertir la información residual, desde un dominio de valores de píxel a un dominio de transformaciones, tal como un dominio de frecuencia. La unidad de procesamiento de transformaciones 52 puede enviar los coeficientes de transformación resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformación para reducir adicionalmente la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes. El grado de cuantificación puede modificarse ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, un escaneado de la matriz que incluye los coeficientes de transformación cuantificados. De

forma alternativa, la unidad de codificación por entropía 56 puede llevar a cabo el escaneado.

[0104] Tras la cuantificación, la unidad de codificación por entropía 56 realiza la codificación por entropía de los coeficientes de transformación cuantificados. Por ejemplo, la unidad de codificación por entropía 56 puede llevar a cabo la codificación de longitud variable adaptativa al contexto (CAVLC), la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación aritmética binaria adaptativa al contexto y basada en sintaxis (SBAC), la codificación por entropía mediante la división en intervalos de probabilidades (PIPE) u otra técnica de codificación por entropía. En el caso de la codificación por entropía basada en el contexto, el contexto puede basarse en bloques contiguos. Tras la codificación por entropía realizada por la unidad de codificación por entropía 56, el flujo de bits codificado puede transmitirse a otro dispositivo (por ejemplo, el descodificador de vídeo 30) o archivar para su posterior transmisión o recuperación.

[0105] La unidad de cuantificación inversa 58 y la unidad de transformación inversa 60 aplican la cuantificación inversa y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia agregando el bloque residual a un bloque predictivo de una de las tramas de la memoria de imágenes de referencia 64. La memoria de imagen de referencia 64 puede implementar o incluir una memoria intermedia de imágenes descodificadas (DPB). La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxeles fraccionarios y usarlos en la estimación de movimiento. El sumador 62 agrega el bloque residual reconstruido al bloque de predicción compensado por movimiento, generado por la unidad de compensación de movimiento 44, para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria de imágenes de referencia 64. El bloque de vídeo reconstruido puede ser usado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 como un bloque de referencia para inter-codificar un bloque en una trama de vídeo posterior.

[0106] El codificador de vídeo 20 representa un ejemplo de un codificador de vídeo que puede configurarse para realizar cualquiera de las técnicas de esta divulgación, solas o en cualquier combinación. Por ejemplo, el codificador de vídeo 20 puede configurarse para determinar si una imagen de capa de referencia tiene habilitado el filtrado a lo ancho de los límites de los elementos, y si es así, inhabilitar la predicción entre capas para una imagen de capa de mejora a partir de la imagen de capa de referencia. Además o de forma alternativa, el codificador de vídeo 20 puede configurarse para codificar una imagen de capa de referencia, y sustancialmente en paralelo, una imagen de capa de mejora que tiene una relación de aspecto mayor, donde el codificador de vídeo 20 puede codificar partes superpuestas de la imagen de capa de mejora usando predicción entre capas y partes no superpuestas de la imagen de capa de mejora sin usar predicción entre capas. Además o de forma alternativa, el codificador de vídeo 20 puede codificar un elemento sintáctico que indica si los límites de los elementos para una capa de mejora están alineados con los límites de los elementos de una imagen de capa de referencia, y codificar la imagen de capa de mejora basándose en el elemento sintáctico. Es decir, el codificador de vídeo 20 puede inhabilitar la predicción entre capas si los límites de los elementos no están alineados, pero habilitar la predicción entre capas si los límites de los elementos están alineados. Además o de forma alternativa, el codificador de vídeo 20 puede codificar un elemento sintáctico para un elemento de capa de mejora que indica un cierto número de elementos de capa de referencia que se tienen que codificar antes de que pueda codificarse el elemento de capa de mejora, y a continuación, tras codificar ese número de elementos de capa de referencia, codificar el elemento de capa de mejora. Cualquiera o todas estas diversas operaciones de codificación pueden realizarse en paralelo con otras operaciones de codificación, con la restricción de que los elementos de capa de referencia utilizados para predicción entre capas para codificar un elemento de capa de mejora se codifican antes de que se codifique el elemento de capa de mejora.

[0107] La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo 30 que puede implementar técnicas para el procesamiento de datos de vídeo en paralelo. En el ejemplo de la FIG. 3, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 70, una unidad de compensación de movimiento 72, una unidad de intra-predicción 74, una unidad de cuantificación inversa 76, una unidad de transformación inversa 78, una memoria de imágenes de referencia 82 y un sumador 80. La memoria de imagen de referencia 82 puede implementar o incluir una memoria intermedia de imágenes descodificadas (DPB). En algunos ejemplos, el descodificador de vídeo 30 puede llevar a cabo un pase de descodificación en general recíproca al pase de codificación descrita con respecto al codificador de vídeo 20 (FIG. 2). La unidad de compensación de movimiento 72 puede generar datos de predicción basados en vectores de movimiento recibidos desde la unidad de descodificación por entropía 70, mientras que la unidad de intra-predicción 74 puede generar datos de predicción basados en indicadores de modo de intra-predicción recibidos desde la unidad de descodificación por entropía 70.

[0108] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado que representa los bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados del codificador de vídeo 20. La unidad de descodificación por entropía 70 del descodificador de vídeo 30 descodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento o indicadores de modo de intra-predicción y otros elementos sintácticos. La unidad de descodificación por entropía 70 remite los vectores de movimiento y otros elementos sintácticos a la unidad de compensación de movimiento 72. El descodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel del fragmento de vídeo y/o el nivel del

bloque de vídeo.

[0109] Cuando el fragmento de vídeo se codifica como un fragmento intra-codificado (I), la unidad de intra-predicción 74 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual, basándose en un modo de intra-predicción indicado, y datos de bloques previamente descodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento inter-codificado (es decir, B, P o GPB), la unidad de compensación de movimiento 72 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual, basándose en los vectores de movimiento y a otros elementos sintácticos recibidos desde la unidad de descodificación por entropía 70. Los bloques predictivos pueden ser generados a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por omisión, basándose en las imágenes de referencia almacenadas en la memoria de imágenes de referencia 82. La unidad de compensación de movimiento 72 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual, analizando los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos del bloque de vídeo actual que está siendo descodificado. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intra-predicción o inter-predicción), usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de inter-predicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia del fragmento, vectores de movimiento para cada bloque de vídeo inter-codificado del fragmento, el estado de inter-predicción para cada bloque de vídeo inter-codificado del fragmento y otra información para descodificar los bloques de vídeo en el fragmento de vídeo actual.

[0110] La unidad de compensación de movimiento 72 también puede realizar la interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo, para calcular valores interpolados para píxeles fraccionarios de bloques de referencia. En este caso, la unidad de compensación de movimiento 72 puede determinar los filtros de interpolación utilizados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos, y utilizar los filtros de interpolación para generar bloques predictivos.

[0111] La unidad de cuantificación inversa 76 cuantifica de manera inversa, es decir, descuantifica, los coeficientes de transformación cuantificados, proporcionados en el flujo de bits y descodificados por la unidad de descodificación por entropía 70. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación QP_Y , calculado por el descodificador de vídeo 30 para cada bloque de vídeo en el fragmento de vídeo para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que debería aplicarse.

[0112] La unidad de transformación inversa 78 aplica una transformación inversa, por ejemplo, una DCT inversa, una transformación inversa entera o un proceso de transformación inversa conceptualmente similar a los coeficientes de transformación, con el fin de producir bloques residuales en el dominio de píxeles.

[0113] Después de que la unidad de compensación de movimiento 72 genera el bloque predictivo para el bloque de vídeo actual, basándose en los vectores de movimiento y otros elementos sintácticos, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales procedentes de la unidad de procesamiento de transformación inversa 88 a los correspondientes bloques predictivos generados por la unidad de compensación de movimiento 72. El sumador 80 representa el componente o los componentes que llevan a cabo esta operación de suma. Si se desea, también puede aplicarse un filtro de desbloqueo para filtrar los bloques descodificados con el fin de eliminar distorsiones de efecto pixelado. También pueden utilizarse otros filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) para allanar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo descodificados en una trama o imagen dada son a continuación almacenados en la memoria de imágenes de referencia 82, que almacena imágenes de referencia usadas para la posterior compensación de movimiento. La memoria de imágenes de referencia 82 almacena también vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1.

[0114] El descodificador de vídeo 30 se puede configurar para realizar cualquiera o todas las técnicas de esta divulgación, solas o en cualquier combinación. Por ejemplo, el descodificador de vídeo 30 puede configurarse para determinar si una imagen de capa de referencia tiene filtrado a lo ancho de los límites de los elementos habilitado, y si es así, inhabilitar la predicción entre capas para una imagen de capa de mejora de la imagen de capa de referencia. Además o de forma alternativa, el descodificador de vídeo 30 puede configurarse para descodificar una imagen de capa de referencia, y sustancialmente en paralelo, una imagen de capa de mejora que tenga una relación de aspecto mayor que la imagen de capa de referencia, donde el descodificador de vídeo 30 puede descodificar partes superpuestas de la imagen de capa de mejora usando la predicción entre capas y partes no superpuestas de la imagen de capa de mejora sin usar la predicción entre capas. Además o de forma alternativa, el descodificador de vídeo 30 puede descodificar un elemento sintáctico que indica si los límites de los elementos para una capa de mejora están alineados con los límites de una imagen de capa de referencia, y descodificar la imagen de capa de mejora basándose en el elemento sintáctico. Es decir, el descodificador de vídeo 30 puede inhabilitar la predicción

entre capas si los límites de los elementos no están alineados, pero habilitar la predicción entre capas si los límites de los elementos están alineados. Además o de forma alternativa, el descodificador de vídeo 30 puede descodificar un elemento sintáctico para un elemento de capa de mejora que indica un cierto número de elementos de capa de referencia a descodificar antes de que se pueda descodificar el elemento de capa de mejora, y a continuación, después de descodificar ese número de elementos de capa de referencia, descodificar el elemento de capa de mejora. Cualquiera o todas estas diversas operaciones de descodificación se pueden realizar en paralelo con otras operaciones de descodificación, con la restricción de que los elementos de capa de referencia utilizados para la predicción entre capas para descodificar un elemento de capa de mejora se descodifiquen antes de descodificar el elemento de capa de mejora.

[0115] La FIG. 4 es un diagrama conceptual que ilustra varias dimensiones escalables para la codificación de vídeo escalable (SVC). La FIG. 4 ilustra un ejemplo de la estructura de escalabilidad de SVC. En particular, se muestra un ejemplo de escalabilidades en diferentes dimensiones en la FIG. 4. En este ejemplo, las escalabilidades se habilitan en tres dimensiones. En la dimensión de tiempo, las frecuencias de tramas con 7,5 Hz, 15 Hz o 30 Hz pueden ser soportadas por la escalabilidad temporal (T). Cuando se soporta escalabilidad espacial (S), se pueden habilitar diferentes resoluciones, como QCIF, CIF y 4CIF. Para cada resolución espacial y frecuencia de tramas específicas, se pueden agregar capas de señal a ruido (SNR) (Q) para mejorar la calidad de la imagen.

[0116] Una vez que el contenido de vídeo se ha codificado de tal manera escalable, se puede usar una herramienta de extracción para adaptar el contenido real entregado de acuerdo con los requisitos de la aplicación, que pueden depender, por ejemplo, de los clientes o el canal de transmisión. En el ejemplo mostrado en la FIG. 4, cada volumen cúbico, es decir, cubo, contiene las imágenes con la misma frecuencia de tramas (nivel temporal), resolución espacial y capas de SNR. Se puede lograr una mejor representación agregando esos cubos (imágenes) en cualquier dimensión. La escalabilidad combinada puede soportarse cuando hay dos, tres o incluso más escalabilidades habilitadas.

[0117] De acuerdo con la especificación SVC, las imágenes con las capas espacial y de calidad más bajas son compatibles con H.264/AVC, y las imágenes en el nivel temporal más bajo forman la capa base temporal, que se puede mejorar con imágenes a niveles temporales más altos. Además de la capa compatible con H.264/AVC, se pueden agregar varias capas de mejora espacial y/o de SNR para proporcionar escalabilidades espacial y/o de calidad. La escalabilidad de SNR también se denomina escalabilidad de calidad. Cada capa de mejora espacial o de SNR puede ser temporalmente escalable, con la misma estructura de escalabilidad temporal que la capa compatible con H.264/AVC. Para una capa de mejora espacial o de SNR, la capa inferior de la que depende también se denomina capa base de esa capa de mejora espacial o de SNR específica.

[0118] La FIG. 5 es un diagrama conceptual que ilustra un ejemplo de la estructura de codificación SVC. En este ejemplo, las imágenes con la capa espacial y de calidad más bajas (imágenes en la capa 0 y capa 1, con resolución QCIF) son compatibles con H.264/AVC. Entre ellas, las imágenes del nivel temporal más bajo forman la capa base temporal, como se muestra en la capa 0 de la FIG. 5. Esta capa base temporal (capa 0) se puede mejorar con imágenes de niveles temporales más altos (capa 1). Además de la capa compatible con H.264/AVC, se pueden agregar varias capas de mejora espacial y/o de SNR para proporcionar escalabilidades espacial y/o de calidad. Por ejemplo, la capa de mejora puede ser una representación CIF con la misma resolución que la capa 2. En este ejemplo, la capa 3 es una capa de mejora de SNR. Como se muestra en este ejemplo, cada capa de mejora espacial o de SNR puede ser temporalmente escalable, con la misma estructura de escalabilidad temporal que la capa compatible con H.264/AVC. Además, una capa de mejora puede mejorar tanto la resolución espacial como la frecuencia de tramas. Por ejemplo, la capa 4 proporciona una capa de mejora de 4CIF, que aumenta aún más la frecuencia de tramas de 15 Hz a 30 Hz.

[0119] La FIG. 6 es un diagrama conceptual que ilustra unidades de acceso (AU) de ejemplo. Cada AU incluye uno o más fragmentos, encapsulados dentro de unidades de capa de abstracción de red (NAL). Puede haber cero o más unidades NAL por unidad de acceso por capa. Un conjunto de unidades NAL que corresponde a una capa dentro de una unidad de acceso se puede denominar un "componente de capa". El ejemplo de la FIG. 6 representa componentes de capa correspondientes a los componentes de capa de la FIG. 5. Como se ilustra en el ejemplo de la FIG. 6, los fragmentos codificados en la misma instancia de tiempo (es decir, dentro de una AU común) son sucesivos en el orden de flujo de bits y forman una unidad de acceso en el contexto de SVC. A continuación, esas unidades de acceso SVC siguen el orden de descodificación, que podría ser diferente del orden de visualización y decidirse, por ejemplo, mediante la relación de predicción temporal.

[0120] Las extensiones escalables de H.264/AVC (Codificación de Vídeo Avanzada) se describen a continuación. Algunas funcionalidades de SVC se heredan de H.264/AVC. En comparación con las normas escalables anteriores, a continuación se revisan algunas de las mayores ventajas de la extensión de SVC a H.264/AVC, a saber, la predicción entre capas y la descodificación de un solo bucle.

[0121] La extensión de SVC de H.264/AVC soporta la descodificación de un solo bucle. Para mantener un descodificador de baja complejidad, la descodificación de un solo bucle es obligatoria en SVC. Con la descodificación de un solo bucle, cada capa soportada se puede descodificar con un solo bucle de compensación de

movimiento. Para lograr esto, el uso de la intra-predicción entre capas solo se permite para macrobloques de capa de mejora, para los cuales la señal de la capa de referencia co-ubicada está intra-codificada. Se requiere además que todas las capas que se utilizan para la predicción entre capas de las capas superiores se codifiquen usando intra-predicción restringida.

[0122] La extensión SVC de H.264/AVC también soporta la predicción entre capas. SVC introduce la predicción entre capas para escalabilidades espaciales y de SNR basadas en textura, residuos y movimiento. La escalabilidad espacial en SVC se ha generalizado para cualquier relación de resolución entre dos capas. La escalabilidad de SNR se puede realizar mediante escalabilidad de granularidad gruesa (CGS) o escalabilidad de granularidad media (MGS). En SVC, dos capas espaciales o de CGS pertenecen a distintas capas de dependencia (indicadas mediante *dependency_id* en la cabecera de la unidad NAL), mientras que dos capas de MGS pueden estar en la misma capa de dependencia. Una capa de dependencia incluye capas de calidad con *quality_id* desde 0 hasta valores más altos, correspondientes a capas de mejora de calidad. En SVC, los procedimientos de predicción entre capas se utilizan para reducir la redundancia entre capas. Se presentan brevemente en los siguientes párrafos.

[0123] El modo de codificación que usa intra-predicción entre capas se denomina modo "IntraBL" en SVC. Para habilitar la descodificación de un solo bucle, solo los macrobloques (MB), que tienen MBs co-ubicados en la capa base codificados como modos de intra restringidos, pueden usar el modo de intra-predicción entre capas. Un modo de intra restringido MB está intra-codificado sin referirse a ninguna muestra de MB inter-codificados contiguos.

[0124] Si se indica que un MB utiliza predicción residual, el MB co-ubicado en la capa base para la predicción entre capas debe ser un MB inter y su residuo se puede muestrear de manera ascendente de acuerdo con la relación de resolución espacial. La diferencia de residuo entre la capa de mejora y la de la capa base está codificada. Es decir, la reconstrucción de la trama actual \hat{I}_e de la capa de mejora es igual a la suma de los coeficientes descuantificados r_e de la capa de mejora, la predicción temporal P_e de la capa de mejora, y los coeficientes residuales normalizados de cuantificación r_b de la capa base.

$$\hat{I}_e = r_e + P_e + r_b \quad (1)$$

[0125] Los vectores de movimiento de la capa base co-ubicados se pueden escalar para generar predictores para los vectores de movimiento de la división de MB o MB en la capa de mejora. Además, hay un tipo de MB llamado modo base, que envía un indicador para cada MB. Si este indicador es verdadero y el MB de capa base correspondiente no es intra, los vectores de movimiento, los modos de división y los índices de referencia se obtienen de la capa base.

[0126] La FIG. 7 es un diagrama conceptual que ilustra elementos de ejemplo de acuerdo con la norma de codificación de video de alta eficiencia (HEVC). HEVC contiene varias propuestas para que el códec sea más compatible con los paralelos, incluidos los elementos y el procesamiento paralelo de frente de onda (WPP). HEVC define los elementos como un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneado de trama de bloques de árbol de codificación del elemento. La división de cada imagen en elementos se denomina división en HEVC. Los elementos en una imagen se ordenan consecutivamente en el escaneado de trama de la imagen como se muestra en la FIG. 7.

[0127] El número de elementos y la ubicación de sus límites se pueden definir para toda la secuencia o cambiar de una imagen a otra. Los límites, de manera similar a los límites de fragmento, rompen las dependencias de análisis y predicción para que un elemento se pueda procesar de forma independiente, pero los filtros en bucle (desbloqueo y SAO) aún pueden cruzar los límites de los elementos. HEVC también especifica algunas restricciones en la relación entre los fragmentos y los elementos.

[0128] Una ventaja de usar elementos es que que no requieren la comunicación entre los procesadores para la descodificación por entropía y la reconstrucción de compensación de movimiento, pero puede ser necesaria la comunicación si *loop_filter_across_tiles_enabled_flag* se establece en 1.

[0129] En comparación con los fragmentos, los elementos tienen típicamente una mejor eficiencia de codificación, porque los elementos permiten formas de división de imágenes que contienen muestras con una correlación potencial mayor que las divisiones, y también porque los elementos reducen la sobrecarga de la cabecera de fragmento.

[0130] El diseño de elementos en HEVC puede proporcionar las siguientes ventajas: 1) habilitar el procesamiento paralelo, y 2) mejorar la eficiencia de codificación al permitir un orden de descodificación cambiado de las unidades de árbol de codificación (CTU) en comparación con el uso de fragmentos, mientras que la ventaja principal es la primera. Cuando se usa un elemento en la codificación de una sola capa, el elemento sintáctico *min_spatial_segmentation_idc* puede ser utilizado por un descodificador para calcular el número máximo de muestras de luma a procesar por un hilo de procesamiento, suponiendo que el descodificador utiliza al máximo la información de descodificación en paralelo. En HEVC, puede haber las mismas interdependencias de imagen entre

los diferentes hilos, por ejemplo, debido a la sincronización de la codificación por entropía o al filtrado de desbloqueo a lo ancho de los límites de los elementos o fragmentos. La norma HEVC incluye una nota que favorece que los codificadores establezcan el valor de `min_spatial_segmentation_idc` en el valor más alto posible.

[0131] La FIG. 8 es un diagrama conceptual que ilustra los frentes de onda para el procesamiento paralelo de frentes de onda (WPP). HEVC define una técnica WPP. Cuando WPP está habilitado, cada fila de CTU de una imagen es una división separada. Sin embargo, en comparación con los fragmentos y los elementos, no se han roto las dependencias de codificación en los límites de filas de CTU. Además, las probabilidades de CABAC se propagan desde la segunda CTU de la fila anterior, para reducir aún más las pérdidas de codificación. Además, WPP no cambia el orden de escaneado de trama regular. Debido a que las dependencias no están rotas, la pérdida de distorsión de velocidad de un flujo de bits WPP es pequeña, en comparación con un flujo de bits no paralelo.

[0132] Cuando se habilita WPP, un cierto número de procesadores hasta el número de filas de CTU puede funcionar en paralelo para procesar la fila (o líneas) de CTU. Sin embargo, las dependencias de frente de onda no permiten que todas las filas de CTU empiecen a descodificarse al comienzo de la imagen. En consecuencia, las filas de CTU tampoco pueden terminar de descodificar al mismo tiempo al final de la imagen. Esto introduce ineficiencias de paralelización que se vuelven más evidentes cuando se usa una gran cantidad de procesadores.

[0133] La FIG. 8 ilustra cómo WPP procesa filas de CTB en paralelo, comenzando cada fila con las probabilidades de CABAC disponibles después de procesar el segundo CTB de la fila anterior.

[0134] De forma similar a H.264/AVC, HEVC también tendrá extensión de codificación de vídeo escalable, actualmente llamada Codificación de Vídeo de Alta Eficiencia Escalable (SHVC), que al menos proporcionará escalabilidad temporal, escalabilidad de SNR y escalabilidad espacial. La convocatoria de propuestas se realizó en julio de 2012. La primera versión del modelo de prueba y borrador de funcionamiento de SHVC estará disponible en marzo de 2013.

[0135] En SHVC, para lograr predicción de textura entre capas, las muestras reconstruidas de capa de referencia se muestrean de forma ascendente en primer lugar cuando la resolución de la capa de referencia es menor que la de la capa de mejora. Incluso en el caso de escalabilidad de SNR, las muestras de la capa de referencia pueden filtrarse antes de ser utilizadas para la predicción entre capas para obtener una mayor eficiencia de codificación. Actualmente, el proceso de muestreo ascendente o de filtrado entre capas se realiza para imágenes de capa completa, que también pueden denominarse componentes de capa o simplemente imágenes.

[0136] En SHVC, se usa la estructura de descodificación de múltiples bucles. Los descodificadores pueden procesar diferentes capas en paralelo. Para facilitar el procesamiento paralelo, F. Henry y S. Pateux, "Procesamiento paralelo de frente de onda", JCTVC-E196, marzo de 2011, propusieron indicar que los límites de los elementos de todas las capas de referencia están alineados con los límites de los elementos de esa capa de mejora. La alineación de los límites de los elementos de las diferentes capas se especifica primero de manera que para dos muestras cualesquiera de imágenes de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa base colocadas también se encontrarán dentro del mismo elemento de capa base.

[0137] En SHVC, se usa la estructura de descodificación de múltiples bucles. Los descodificadores pueden procesar diferentes capas en paralelo. Para facilitar el procesamiento paralelo, se ha propuesto indicar que los límites de los elementos de todas las capas de referencia están alineados con los límites de los elementos de esa capa de mejora. La sintaxis y la semántica de la alineación de los límites de los elementos y la alineación de las diferentes capas se especifica en la Tabla 7 siguiente, que ilustra las modificaciones de una tabla descrita en K. Sühling, R. Skupin, T. Schierl, Indicación de la alineación de los límites de los elementos, JCTVC-L0197, enero de 2013 y JCTVC-M0464, mencionado anteriormente, donde el texto *en cursiva* representa las adiciones propuestas.

TABLA 7
(A Continuación)

TABLA 7

vui_parameters() {	Descriptor
aspect_ratio_info_present_flag	u(1)
...	
if(bitstream_restriction_flag) {	
tiles_fixed_structure_flag	u(1)
<i>if (nuh_layer_id > 0)</i>	
tile_boundaries_aligned_flag	u(1)
motion_vectors_over_pic_boundaries_flag	u(1)
restricted_ref_pic_lists_flag	u(1)
min_spatial_segmentation_idc	u(8)
max_bytes_per_pic_denom	ue(v)
max_bits_per_minicu_denom	ue(v)
log2_max_mv_length_horizontal	ue(v)
log2_max_mv_length_vertical	ue(v)
}	
}	

5 **[0138]** La semántica para el elemento sintáctico "tile_boundaries_aligned_flag" puede ser la siguiente: tile_boundaries_aligned_flag igual a 1 indica que todos los límites de los elementos de la imagen de capa base correspondiente tienen límites de los elementos correspondientes en la capa de mejora dada y no existe ningún límite de elemento adicional en el capa de mejora. Por consiguiente, para dos muestras cualesquiera de imágenes de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa base colocadas también estarán dentro del mismo elemento de capa de referencia, y para dos muestras cualesquiera de imagen de capa de referencia que se encuentren dentro del mismo elemento de capa de referencia. las muestras de capa de mejora colocadas también se encontrarán dentro del mismo elemento de capa de mejora. El elemento sintáctico tile_boundaries_aligned_flag igual a 0 indica que no hay restricciones en la configuración del elemento entre la capa de mejora correspondiente y la capa de referencia dada.

15 **[0139]** La FIG. 9 es un diagrama conceptual que ilustra un ejemplo de procesamiento paralelo en un codificador SHVC, por ejemplo, un codificador de vídeo 20 o un descodificador de vídeo 30. En SHVC, se puede predecir una capa de mejora a partir de una capa de referencia, tal como una capa base o una capa de mejora inferior. Además, una o ambas imágenes de la capa de referencia y la capa de mejora se pueden dividir en elementos. Como se ilustra en el ejemplo de la FIG. 9, la imagen en la capa de referencia se divide en dos elementos, B0 y B1, mientras que la imagen en la capa de mejora también se divide en dos elementos, E0 y E1. Aunque se utiliza el mismo número de elementos y las mismas posiciones relativas de los elementos para la imagen de capa de mejora y la imagen de capa de referencia en este ejemplo, debe entenderse que, en general, una imagen de capa de mejora puede incluir el mismo o diferentes números de elementos como una imagen de capa de referencia. Además, en algunos ejemplos, se puede predecir una imagen de capa de mejora a partir de dos o más imágenes de capa de referencia, cualquiera o todas las cuales pueden tener el mismo o diferente número de elementos.

30 **[0140]** Como se describe con mayor detalle a continuación, el codificador de vídeo 20 y el descodificador de vídeo 30 puede estar configurado para utilizar diversas restricciones cuando se codifica una imagen de capa de mejora usando una imagen de capa de referencia para la referencia, y una o ambas de los cuales se dividen en los elementos. Las FIGS. 10-14, explicadas con mayor detalle a continuación, son representativas de algunas de las diversas restricciones con las que se puede configurar el codificador de vídeo 20 y el descodificador de vídeo 30.

35 **[0141]** La FIG. 10 es un diagrama conceptual que ilustra una imagen de capa de mejora de ejemplo 100 predicha a partir de una imagen de capa de referencia 104. En este ejemplo, la imagen de capa de mejora 100 incluye los elementos 102A-102D, mientras que la imagen de capa de referencia 104 incluye los elementos 106A, 106B. Como se indica mediante las líneas discontinuas de la FIG. 10, el elemento 102B de la imagen de capa de mejora 100 corresponde al elemento 106A de la imagen de capa de referencia 104, mientras que el elemento 102C de la imagen de capa de mejora 100 corresponde al elemento 106B de la imagen de capa de referencia 104.

40 **[0142]** En este ejemplo, la imagen de capa de mejora 100 tiene una relación de aspecto diferente a la imagen de capa de referencia 104. Por ejemplo, la imagen de capa de referencia 104 puede tener una relación de aspecto 4:3, mientras que la imagen de capa de mejora 100 puede tener una relación de aspecto de 16:9. De este modo, los elementos 102A, 102D de la imagen de capa de mejora 100 no tienen elementos correspondientes en la imagen de

capa de referencia 104. Por ejemplo, el elemento 102A de la imagen de capa de mejora 100 incluye la muestra 108. Tal como se indica mediante sombreado vertical, la muestra 108 no tiene una muestra de capa de referencia colocada (RL) disponible en la imagen de capa de referencia 104. De forma similar, la muestra 114 del elemento 102D de la imagen de capa de mejora 100 no tiene una muestra de RL colocada disponible. Sin embargo, las muestras 110, 112 tienen muestras de capa de referencia colocadas disponibles en la imagen de capa de referencia 104 (como se indica mediante sombreado cruzado). En particular, las muestras 110, 112 corresponden a las muestras 116, 118 de la imagen de capa de referencia 104.

[0143] De esta manera, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos de vídeo de los elementos 102B, 102C usando predicción entre capas con respecto a los elementos 106A, 106B, respectivamente. Sin embargo, debido a que los elementos 102A, 102D no tienen partes correspondientes en la imagen de capa de referencia 104, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar los elementos 102A, 102D sin usar predicción entre capas, por ejemplo, usando intra-predicción o predicción temporal. La codificación de la imagen de capa de mejora 100 puede producirse sustancialmente en paralelo con la codificación de la imagen de capa de referencia 104. Por ejemplo, un codificador de vídeo (por ejemplo, codificador de vídeo 20 o descodificador de vídeo 30) puede codificar el elemento 106A, a continuación codificar el elemento de código 102B sustancialmente en paralelo con el elemento 106B. Además, la codificación de los elementos 102A, 102D puede ocurrir antes, sustancialmente en paralelo con, o después de la codificación de los elementos 106A, 106B, porque los elementos 102A, 102D no dependen de los datos de la imagen de capa de referencia 104.

[0144] La FIG. 10 ilustra un ejemplo en el que se puede decir que los límites de los elementos 102B, 102C están alineados con los límites de los elementos 106A, 106B. En algunos ejemplos, puede decirse que los límites de los elementos están alineados cuando, para dos muestras cualesquiera de imágenes de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa de referencia colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de referencia y para dos muestras cualesquiera de imágenes de capa de referencia que se encuentran dentro del mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de mejora. Dado que dos muestras cualesquiera dentro del elemento 102A corresponderían a las muestras colocadas dentro del elemento 106A, y asimismo, dos muestras cualesquiera dentro del elemento 106A corresponderían a las muestras colocadas dentro del elemento 102A, se puede decir que los límites del elemento 102A están alineados con los límites del elemento 106A. De manera similar, debido a que dos muestras cualesquiera dentro del elemento 102B corresponderían a muestras colocadas dentro del elemento 106B, y de la misma manera, dos muestras cualesquiera dentro del elemento 106B corresponderían a muestras colocadas dentro del elemento 102B, se podría decir que los límites del elemento 102B están alineados con los límites del elemento 106B.

[0145] De esta manera, la FIG. 10 representa un ejemplo en el que el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar (por ejemplo, codificar o descodificar) una imagen de capa base que comprende una primera relación de aspecto y codificar, sustancialmente en paralelo con la imagen de capa base, una capa de imagen de mejora que comprende una segunda relación de aspecto, en la que la segunda relación de aspecto es mayor que la primera relación de aspecto y en la que la codificación de la imagen de capa de mejora comprende partes de codificación de la imagen de capa de mejora que están fuera de un área definida por la primera relación de aspecto sin utilizar predicción de capa

[0146] La FIG. 11 es un diagrama conceptual que ilustra un ejemplo de escalabilidad espacial extendida en el que algunas muestras de capa de referencia (o capa base) no tienen muestras de capa de mejora correspondientes. En particular, el ejemplo de la FIG. 11 ilustra la imagen de capa de mejora 130 y la imagen de capa de referencia 136. Como se muestra en este ejemplo, cuando la imagen de capa de referencia 136 se muestrea de forma ascendente espacialmente para usar como una imagen de referencia para codificar la imagen de capa de mejora 130, hay regiones de la imagen de capa de referencia 136 que no corresponden a partes de la imagen de capa de referencia 130.

[0147] Más particularmente, la imagen de capa de referencia 136 incluye dos elementos, el elemento 138A y el elemento 138B, mientras que la imagen de capa de mejora 130 incluye dos elementos en general correspondientes, el elemento 132A y el elemento 132B. Las líneas de puntos discontinuas dentro de la imagen de capa de referencia 136 definen regiones 140A, 140B que corresponden a los elementos 132A, 132B, respectivamente. Sin embargo, las regiones 142A, 142B entre las líneas de puntos discontinuas y los límites exteriores de la imagen de capa de referencia 136 definen regiones que no corresponden a las regiones reales de la imagen de capa de mejora 130. Es decir, las líneas de puntos discontinuas alrededor de los límites de la imagen de capa de mejora 130 definen una región 134 entre los límites exteriores de la imagen de capa de mejora 130 y las líneas de puntos discontinuas que no incluye muestras para la imagen de capa de mejora 130, pero corresponde a regiones 142A, 142B.

[0148] Por ejemplo, la FIG. 11 ilustra dos muestras de ejemplo para el elemento 138A de la imagen de capa de referencia 136: muestra 144 y muestra 148. Debido a que la muestra 144 está dentro de los límites de la región 140A, puede decirse que la muestra 144 corresponde a la muestra 152 del elemento 132A en la imagen de capa de mejora 130. Por otro lado, la muestra 148 no corresponde a una muestra de la imagen de capa de mejora 130; en

cambio, la región 142A corresponde a una parte dentro de la región 134 en la que no hay muestras presentes. De esta manera, la muestra 148 se puede describir como un ejemplo de una muestra que no tiene una muestra de capa de mejora correspondiente.

[0149] De forma similar, la FIG. 11 ilustra dos muestras de ejemplo para el elemento 138B de la imagen de capa de referencia 136: muestra 146 y muestra 150. Debido a que la muestra 146 está dentro de los límites de la región 140A, puede decirse que la muestra 146 corresponde a la muestra 154 del elemento 132B en la imagen de capa de mejora 130. Por otro lado, la muestra 150 no corresponde a una muestra de la imagen de capa de mejora 130; en cambio, la región 142B corresponde a una parte dentro de la región 134 en la que no hay muestras presentes. De esta manera, la muestra 150 se puede describir como otro ejemplo de una muestra que no tiene una muestra de capa de mejora correspondiente.

[0150] De esta manera, la FIG. 11 proporciona un ejemplo en el que los elementos 138A, 138B están alineados con los elementos 132A, 132B, respectivamente. Es decir, como se analizó anteriormente, puede decirse que los elementos están "alineados" cuando, para dos muestras cualesquiera de imágenes de capa de mejora que se encuentren dentro del mismo elemento de capa de mejora, las muestras de capa de referencia colocadas, si están disponibles, también deben estar dentro del mismo elemento de capa de referencia y para dos muestras cualesquiera de imagen de capa de referencia que se encuentren dentro del mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también se encontrarán dentro del mismo elemento de capa de mejora. Es decir, aunque los límites de los elementos 138A, 138B no están estrictamente alineados con los límites de los elementos 132A, 132B, respectivamente, las muestras 148, 150 de la imagen de capa de referencia 136 no corresponden a las muestras de capa de mejora disponibles. Todas las muestras dentro del elemento 132A tienen muestras correspondientes dentro de la región 140A, y del mismo modo, todas las muestras dentro del elemento 132B tienen muestras correspondientes dentro de la región 140B. De forma similar, todas las muestras dentro de la región 140A corresponden a muestras dentro del elemento 132A, y todas las muestras dentro de la región 140B corresponden a muestras dentro del elemento 132B. Por lo tanto, se puede decir que los elementos 132A, 132B están alineados con los elementos 138A, 138B, respectivamente.

[0151] La FIG. 12 es un diagrama conceptual que ilustra una imagen de capa de referencia 162 y una imagen de capa de mejora 160. La imagen de capa de mejora 160 y la imagen de capa de referencia 162 pueden ocurrir en la misma unidad de acceso (o en caso contrario corresponderse entre sí). La imagen de capa de mejora 160 se divide en elementos 164A, 164B y 164C, mientras que la imagen de capa de referencia 162 se divide en elementos 168A, 168B, 168C. Además, la FIG. 12 ilustra bloques de ejemplo 166, 170 en elementos 164B, 168B, respectivamente. Puede decirse que el bloque 166 corresponde al bloque 170, en ese bloque 166 está colocado junto con el bloque 170.

[0152] De acuerdo con ciertas técnicas de ejemplo de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar configurados para impedir o permitir la predicción entre capas del bloque 166 desde el bloque 170 basándose en diversas condiciones. En un ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para evitar la codificación del bloque 166 usando predicción de textura entre capas y/o predicción de sintaxis entre capas desde el bloque 170 cuando se filtra entre capas y/o el filtrado de capa de referencia a través de los elementos en la imagen de capa de referencia 162 está habilitado. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos que indican si los límites de las distintas capas, como la capa de mejora y la capa de referencia, están alineados e indican si al menos una de predicción de textura entre capas y predicción de sintaxis entre capas son está permitida a lo largo ni a lo ancho de los límites de los elementos cuando el filtrado entre capas o el filtrado de capa de referencia a lo ancho de los límites de los elementos está habilitado para el bloque 170.

[0153] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden, además o de forma alternativa, codificar datos que indican si los límites de los elementos entre diferentes capas (por ejemplo, los límites de los elementos 164, 168) están alineados, así como datos que indican si la predicción de textura entre capas y/o la predicción de sintaxis entre capas no están permitidas a lo largo o a lo ancho de los límites de los elementos. Cuando la predicción de textura entre capas y la predicción de sintaxis entre capas no están permitidas a lo largo ni a lo ancho de los límites de los elementos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar bloques de un elemento sin codificar elementos de capa base no alineados. Por ejemplo, si el bloque 177 (colocado con el bloque 160) estuviera en un elemento que no estuviera alineado con el bloque 164B, el codificador de vídeo 20 y el descodificador de vídeo 30 podrían codificar el bloque 160 sin el primer bloque de codificación 177 y, por lo tanto, podrían codificar el bloque 160 en paralelo con el bloque 177.

[0154] Los datos que indican si la predicción de textura entre capas y la predicción de sintaxis entre capas no están permitidas a lo largo ni a lo ancho de los límites de los elementos puede comprender un elemento sintáctico `inter_layerpred_not_along_tile_boundary_flag`, como se ha explicado anteriormente con respecto a la Tabla 1. Además o de forma alternativa, los datos que indican si la predicción de textura entre capas y la predicción de sintaxis entre capas no están permitidas a lo largo o a lo ancho de los límites de los elementos pueden codificarse como información de usabilidad de vídeo (VUI) y/o como parte de un mensaje de información de mejora suplementaria (SEI).

[0155] La FIG. 13 es un diagrama conceptual que ilustra imágenes de ejemplo que tienen diferentes relaciones de aspecto. En particular, en este ejemplo, la imagen de capa de mejora 180 tiene una relación de aspecto diferente a la imagen de capa de referencia 182. Por ejemplo, la imagen de capa de mejora 180 puede tener una relación de aspecto de 16x9, mientras que la imagen de capa de referencia 182 puede tener una relación de aspecto de 4:3. Una relación de aspecto de 16x9 es un ejemplo de una relación de aspecto que es mayor que la relación de aspecto de 4:3. En este ejemplo, la imagen de capa de mejora 180 se divide en cuatro elementos, concretamente, los elementos 184A-184D. La imagen de capa de referencia 182 está dividida en elementos 186A, 186B. En general, puede decirse que el elemento 184B corresponde al elemento 186A, mientras que el elemento 184C se puede decir que corresponde al elemento 186B.

[0156] Aunque cada uno de los elementos 184A-184D y 186A, 186B incluye una pluralidad de muestras, solo se muestran ciertas muestras con fines de brevedad y análisis. En particular, en este ejemplo, las muestras 188 se ilustran dentro del elemento 184B, las muestras 192 se ilustran dentro del elemento 186A, y las muestras 190 se ilustran dentro del elemento 184D. Además, en este ejemplo, las muestras 188 corresponden a las muestras 192. Es decir, suponiendo que el elemento 184B tenga la misma resolución espacial que el elemento 186A (originalmente o después del muestreo espacial ascendente), las muestras 188 se colocan con las muestras 192. Además, como se muestra en la FIG. 13, las muestras 188 están dentro del mismo elemento de capa de mejora, el elemento 184B, y las muestras 192 están dentro del mismo elemento de capa de referencia, elemento 186A. De esta manera, las muestras 188 representan ejemplos de dos muestras de imagen de capa de mejora que se encuentran dentro del mismo elemento de capa de mejora, y las muestras 192 representan ejemplos de muestras de capa de referencia colocadas disponibles. Lo contrario también es cierto: las muestras 192 representan ejemplos de dos muestras de capa de referencia, dentro del mismo elemento de capa de referencia, para las cuales están disponibles muestras de capa de mejora colocadas (muestras 188) y también en el mismo elemento de capa de mejora (es decir, elemento 184B). Esta propiedad es verdadera para cualquier conjunto de par de muestras dentro del elemento 184B y muestras colocadas dentro del elemento 186A.

[0157] Por consiguiente, se puede decir que los elementos 184B y 186A tienen límites de elementos alineados. Por lo tanto, un codificador de vídeo, como el codificador de vídeo 20 o el decodificador de vídeo 30, puede realizar predicción entre capas, usando muestras 192 como muestras de referencia para codificar (por ejemplo, codificación mediante el codificador de vídeo 20 o decodificación mediante el decodificador de vídeo 30) las muestras 188. Además, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar el elemento 184B sustancialmente en paralelo con el elemento 186B, suponiendo que el codificador de vídeo codificó el elemento 186A antes de codificar el elemento 184B, y también que el filtrado no se realiza a lo ancho de los límites de la capa de referencia 182, por ejemplo a lo ancho del límite entre los elementos 186A y 186B.

[0158] Las muestras 190, en este ejemplo, se muestran en el elemento 184D. Sin embargo, debido a que la imagen de capa de mejora 180 tiene una relación de aspecto diferente a la imagen de capa de referencia 182, no hay ningún elemento que corresponda al elemento 184 en la imagen de capa de referencia 182. Como tal, no hay muestras disponibles que correspondan (o estén ubicadas con) las muestras 190 en la imagen de capa de referencia 182. Aunque no hay muestras correspondientes de la imagen de capa de referencia 182 a las muestras 190 de la imagen de capa de mejora 180, (ni tampoco ningún elemento correspondiente al elemento 184D) los límites de elemento de los elementos de la imagen de capa de mejora 180 pueden definirse sin embargo como alineados con los límites de elemento de los elementos de la imagen de capa de referencia 182.

[0159] En consecuencia, cuando los límites de los elementos de la imagen de capa de mejora 180 y la imagen de capa de referencia 182 están alineados (como en el ejemplo de la FIG. 13) y cuando la predicción de textura entre capas y el filtrado de capa de referencia no se realizan a lo ancho de los límites de elemento de la imagen de capa de referencia 182, el codificador de vídeo 20 puede codificar un valor que indica esto para un elemento sintáctico, por ejemplo, para `inter_layerpred_not_along_tile_boundary_flag`. Del mismo modo, el decodificador de vídeo 30 puede decodificar un valor para un elemento sintáctico, como `inter_layerpred_not_along_tile_boundary_flag`, indicando que los límites de los elementos de la imagen de capa de mejora 180 y la imagen de capa de referencia 182 están alineados y que la predicción de textura entre capas y el filtrado de capa de referencia no se realizan a lo ancho de los límites de los elementos de la imagen de capa de referencia 182.

[0160] En consecuencia, el codificador de vídeo 20 y el decodificador de vídeo 30 representan ejemplos de un codificador de vídeo configurado para codificar (por ejemplo, codificar o decodificar) un valor para un elemento sintáctico representativo de si dos muestras cualesquiera de capa de referencia, colocadas con dos muestras de imagen de capa de mejora respectivas dentro de un elemento de capa de mejora común, deben estar dentro de un elemento de capa de referencia común, y codificar las muestras de imagen de capa de mejora basándose, al menos en parte, en el valor del elemento sintáctico.

[0161] La FIG. 14 es un diagrama conceptual que ilustra una imagen de capa de mejora 200 y una imagen de capa de referencia 202 que tienen diferentes números de elementos. En este ejemplo, la imagen de capa de mejora 200 incluye ocho elementos, elementos 204A-204H, mientras que la imagen de capa de referencia 202 incluye seis elementos, elementos 206A-206F. En consecuencia, los límites de los elementos de capa de mejora 204A-204H no

están alineados con los límites de los elementos de capa de referencia 206A-206F. Es decir, hay al menos dos muestras de al menos uno de los elementos 204A-204H para las cuales las muestras colocadas de la imagen de capa de referencia 202 no están en el mismo elemento de capa de referencia 206A-206F. Asimismo, hay al menos dos muestras de al menos uno de los elementos 206A-206F para las cuales las muestras colocadas de la imagen de capa de mejora 200 no están en el mismo elemento de capa de mejora 204A-204H.

[0162] Sin embargo, incluso sin alineación de límites entre los elementos de la imagen de capa de mejora 200 y capa de imagen de referencia 202, un codificador de vídeo (como codificador de vídeo 20 o descodificador de vídeo 30) puede todavía codificar uno o más de los elementos de capa de mejora 204A-204H sustancialmente en paralelo con uno o más de los elementos de capa de referencia 206A-206F, usando técnicas de esta divulgación.

[0163] En particular, de acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar, para cualquiera o todos los elementos 204A-204H, un valor para un elemento sintáctico, tal como **max_ref_tile_dec_idc_minus1[i]** (donde corresponde al $i^{\text{ésimo}}$ elemento de los elementos 204A-204H), que indica un cierto número de elementos en la imagen de capa de referencia 202 (que es una imagen de capa base, relativa a la imagen de capa de mejora 200) que debe codificarse antes de que el elemento correspondiente de la imagen de capa de mejora 200 pueda codificarse. En particular, cuando el valor se expresa en forma de **max_ref_tile_dec_idc_minus1[i]**, el valor de **max_ref_tile_dec_idc_minus1[i]** es uno menos que el número de elementos en la imagen de capa de referencia 202 que se codificará antes de que se pueda codificar el correspondiente elemento de la imagen de capa de mejora 200; de este modo, el descodificador de vídeo 30 puede agregar uno al valor de **max_ref_tile_dec_idc_minus1[i]** para determinar el número real de elementos en la imagen de capa de referencia 202 que se tienen que codificar.

[0164] Por ejemplo, en el ejemplo de la FIG. 14, el codificador de vídeo 20 puede codificar valores para **max_ref_tile_dec_idc_minus1[i]** de acuerdo con la Tabla 8 a continuación:

TABLA 8

Elem	<i>i</i>	max_ref_tile_dec_idc_minus1[i]
204A	0	0
204B	1	1
204C	2	2
204D	3	2
204E	4	3
204F	5	4
204G	6	5
204H	7	5

[0165] En este ejemplo, el codificador de vídeo 20 puede codificar un valor de 0 para **max_ref_tile_dec_idc_minus1[0]**, que corresponde al elemento 204A, porque al sumar 1 al valor de 0 se obtiene el valor 1. Este valor, a su vez, indica que 1 elemento de la imagen de capa de referencia 202 (concretamente, el elemento 206A) debe codificarse antes de que el elemento 204A pueda codificarse. Como puede apreciarse en la FIG. 14, codificando el elemento 206A, todas las muestras de la imagen de capa de referencia 202 correspondientes al elemento 204A se habrán codificado después del elemento de codificación 206A, y por lo tanto, el codificador de vídeo 20 puede indicar que solo el elemento 206A debe codificarse antes de que el elemento 204A pueda codificarse. Del mismo modo, el descodificador de vídeo 30 puede usar el valor de **max_ref_tile_dec_idc_minus1[0]** para determinar que después de que se haya codificado el elemento 206A, el elemento 204A se puede codificar, por ejemplo, sustancialmente en paralelo (o en paralelo) con cualquiera de los elementos 206B-206F.

[0166] También en este ejemplo, el codificador de vídeo 20 puede codificar un valor de 1 para **max_ref_tile_dec_idc_minus1[1]**, que corresponde al elemento 204B, porque al agregar 1 al valor de 1 se obtiene el valor 2. Este valor, a su vez, indica que se tienen que codificar 2 elementos de la imagen de capa de referencia 202 (a saber, los elementos 206A, 206B) antes de que se pueda codificar el elemento 204B. Como puede apreciarse en la FIG. 14, codificando los elementos 206A, 206B, todas las muestras de la imagen de capa de referencia 202 correspondiente al elemento 204B se habrán codificado después de codificar el elemento 206B, y por lo tanto, el codificador de vídeo 20 puede indicar que los elementos 206A, 206B tienen que codificarse antes de que pueda codificarse el elemento 204B. Del mismo modo, el descodificador de vídeo 30 puede usar el valor de **max_ref_tile_dec_idc_minus1[1]** para determinar que después de que se haya codificado el elemento 206B, se

puede codificar el elemento 204B, por ejemplo, sustancialmente en paralelo (o en paralelo) con cualquiera de los elementos 206C-206F.

[0167] Además, en este ejemplo, el codificador de vídeo 20 puede codificar un valor de 2 para **max_ref_tile_dec_idc_minus1**[2], que corresponde al elemento 204C, porque al sumar 1 al valor de 2 se obtiene el valor 3. Este valor, a su vez, indica que se tienen que codificar 3 elementos de la imagen de capa de referencia 202 (concretamente, los elementos 206A-206C) antes de que se pueda codificar el elemento 204C. De manera similar, el codificador de vídeo 20 puede codificar un valor de 2 para **max_ref_tile_dec_idc_minus1**[3], correspondiente al elemento 204D. Como puede apreciarse en la FIG. 14, codificando los elementos 206A-206C, todas las muestras de la imagen de capa de referencia 202 correspondiente a los elementos 204C y 204D se habrán codificado después del elemento de codificación 206C, y por lo tanto, el codificador de vídeo 20 puede indicar que los elementos 206A-206C se tienen que codificar antes que se puedan codificar los elementos 204C, 204D. Del mismo modo, el descodificador de vídeo 30 puede usar los valores de **max_ref_tile_dec_idc_minus1**[2] y **max_ref_tile_dec_idc_minus1**[3] para determinar que después de que el elemento 206C se haya codificado, pueden codificarse los elementos 204C y 204D, por ejemplo, sustancialmente en paralelo (o en paralelo) con cualquiera de los elementos 206D-206F.

[0168] Además, en este ejemplo, el codificador de vídeo 20 puede codificar un valor de 3 para **max_ref_tile_dec_idc_minus1**[4], que corresponde al elemento 204E, porque al sumar 1 al valor de 3 se obtiene el valor 4. Este valor, a su vez, indica que se tienen que codificar 4 elementos de la imagen de capa de referencia 202 (a saber, los elementos 206A, 206B) antes de que se pueda codificar el elemento 204E. Como puede apreciarse en la FIG. 14, codificando los elementos 206A-206D, todas las muestras de la imagen de capa de referencia 202 correspondiente al elemento 204E se habrán codificado después del elemento de codificación 206D, y por lo tanto, el codificador de vídeo 20 puede indicar que los elementos 206A-206D deben codificarse antes de que pueda codificarse el elemento 204E. Del mismo modo, el descodificador de vídeo 30 puede usar el valor de **max_ref_tile_dec_idc_minus1**[4] para determinar que después de que el elemento 206D se haya codificado, se puede codificar el elemento 204E, por ejemplo, sustancialmente en paralelo (o en paralelo) con cualquiera de los elementos 206E, 206F.

[0169] También en este ejemplo, el codificador de vídeo 20 puede codificar un valor de 4 para **max_ref_tile_dec_idc_minus1**[5], que corresponde al elemento 204F, porque al agregar 1 al valor de 4 se obtiene el valor 5. Este valor, a su vez, indica que se tienen que codificar 5 elementos de la imagen de capa de referencia 202 (a saber, los elementos 206A-206E) antes de que se pueda codificar el elemento 204F. Como puede apreciarse en la FIG. 14, codificando los elementos 206A-206E, todas las muestras de la imagen de capa de referencia 202 correspondiente al elemento 204F se habrán codificado después del elemento de codificación 206E, y por lo tanto, el codificador de vídeo 20 puede indicar que los elementos 206A-206E deben codificarse antes de que pueda codificarse el elemento 204F. Del mismo modo, el descodificador de vídeo 30 puede usar el valor de **max_ref_tile_dec_idc_minus1**[5] para determinar que después de que se haya codificado el elemento 206E, se puede codificar el elemento 204F, por ejemplo, sustancialmente en paralelo (o en paralelo) con el elemento 206F.

[0170] Además, en este ejemplo, el codificador de vídeo 20 puede codificar un valor de 5 para **max_ref_tile_dec_idc_minus1**[6], que corresponde al elemento 204G, porque al agregar 1 al valor de 5 se obtiene el valor 6. Este valor, a su vez, indica que los 6 elementos de la imagen de capa de referencia 202 (a saber, los elementos 206A-206F) se tienen que codificar antes de poder codificar el elemento 204G. De manera similar, el codificador de vídeo 20 puede codificar un valor de 5 para **max_ref_tile_dec_idc_minus1**[7], que corresponde al elemento 204H. Como puede apreciarse en la FIG. 14, codificando los elementos 206A-206F, todas las muestras de la imagen de capa de referencia 202 correspondiente a los elementos 204G y 204H se habrán codificado después del elemento de codificación 206F, y por lo tanto, el codificador de vídeo 20 puede indicar que los elementos 206A-206F deben codificarse antes de que se puedan codificar los elementos 204G, 204H. Del mismo modo, el descodificador de vídeo 30 puede usar los valores de **max_ref_tile_dec_idc_minus1**[6] y **max_ref_tile_dec_idc_minus1**[7] para determinar que después de codificar el elemento 206F, pueden codificarse los elementos 204G y 204H, por ejemplo, sustancialmente en paralelo (o en paralelo) con los elementos de una imagen posterior en la capa de referencia (es decir, una imagen en la capa de referencia que sigue a la imagen 202 en orden de codificación).

[0171] En consecuencia, el codificador de vídeo 20 y el descodificador de vídeo 30 representan ejemplos de un codificador de vídeo configurado para codificar, para un elemento de una imagen de capa de mejora, datos que indican un número de elementos en una imagen de capa base que deben codificarse antes de que pueda codificarse el elemento de la imagen de capa de mejora, con los elementos de código de la imagen de capa base que corresponden al número de elementos y, después de codificar los elementos de la imagen de capa base, codificar el elemento de la imagen de capa de mejora sustancialmente en paralelo con al menos otro elemento en la imagen de capa base.

[0172] La FIG. 15 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la habilitación o inhabilitación de predicción entre capas de acuerdo con las técnicas de esta divulgación. El procedimiento de la FIG. 15 en general se describe como realizado por un codificador de vídeo. El codificador de vídeo puede corresponder,

por ejemplo, al codificador de vídeo 20 o al descodificador de vídeo 30, u otros dispositivos codificadores de vídeo (por ejemplo, dispositivos de transcodificación de vídeo).

[0173] Inicialmente, el codificador de vídeo puede codificar uno o más elementos de capa de referencia contiguos (250). Es decir, el codificador de vídeo puede codificar bloques (por ejemplo, CU o PU) de uno o más elementos en una imagen de capa de referencia. El codificador de vídeo puede entonces determinar si el filtrado a lo ancho de los límites de los elementos está habilitado (252). Por ejemplo, cuando se realiza mediante el codificador de vídeo 20, el codificador de vídeo 20 puede determinar si se habilita el filtrado a lo ancho de los límites de los elementos para la imagen de capa de referencia. El codificador de vídeo 20 también puede codificar un elemento sintáctico que indica si está habilitado el filtrado a lo ancho de los límites de los elementos para la imagen de capa de referencia. Por otra parte, cuando se realiza mediante el descodificador de vídeo 30, el descodificador de vídeo 30 puede determinar si el filtrado a lo ancho de los límites de los elementos se habilita desde un elemento sintáctico.

[0174] El filtrado a lo ancho de los límites de los elementos puede incluir filtrado entre capas a lo ancho de los límites de los elementos (por ejemplo, cuando la imagen de capa de referencia se codifica con referencia a otra imagen de capa de referencia para la cual está habilitado el filtrado a lo ancho de los límites de los elementos) y/o filtrado de capa de referencia, por ejemplo, desbloqueo, muestreo ascendente de la imagen completa, o cualquier otra operación de filtrado que use un filtro aplicado a píxeles (también denominados muestras) de dos o más elementos distintos de la imagen de capa de referencia.

[0175] En algunos ejemplos, el codificador de vídeo puede determinar además si los límites de los elementos de las diferentes capas están alineados. Por ejemplo, el codificador de vídeo 20 puede determinar si alinear los límites de los elementos y codificar un elemento sintáctico que indique si los límites de los elementos están alineados, mientras que el descodificador 30 puede descodificar el elemento sintáctico para determinar si los límites de los elementos están alineados. Del mismo modo, el codificador de vídeo puede codificar un elemento sintáctico que indica si la predicción entre capas no está permitida a lo largo o a lo ancho de los límites de los elementos, por ejemplo, un `inter_layerpred_not_along_tile_boundary_flag`, como se analizó anteriormente con respecto a la Tabla 1. Además o de forma alternativa, tales elementos sintácticos se pueden incluir en cualquiera o en la totalidad de la información de usabilidad de vídeo (VUI), un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS), un conjunto de parámetros de vídeo (VPS), una cabecera de fragmento, un mensaje de información de mejora suplementaria (SEI) o similar.

[0176] Si se habilita el filtrado a lo ancho de los límites de los elementos para la imagen de capa de referencia (rama "Sí" de 252), el codificador de vídeo puede inhabilitar la predicción entre capas para una imagen de capa de mejora (254). El codificador de vídeo puede entonces codificar bloques de la imagen de capa de mejora sin usar la predicción entre capas (256). La predicción entre capas puede incluir una o ambas predicciones de textura entre capas (por ejemplo, predicción de datos de luminancia y/o crominancia) y predicción de sintaxis entre capas. El codificador de vídeo puede codificar bloques de la imagen de capa de mejora en paralelo con cualquiera de los bloques de la imagen de capa de referencia, porque la imagen de capa de referencia no se usará realmente como referencia cuando se codifique la imagen de capa de mejora. De este modo, aunque el ejemplo de la FIG. 15 representa la codificación de elementos de capa de referencia contiguos antes de la codificación de los bloques de la imagen de capa de mejora, en algunos ejemplos, el codificador de vídeo puede simplemente determinar si el filtrado a lo ancho de los límites de los elementos está habilitado para la imagen de capa de referencia y, de ser así, comenzar inmediatamente la codificación de la imagen de capa de mejora en paralelo con la imagen de capa de referencia.

[0177] Por otro lado, si el filtrado a lo ancho de los límites de los elementos para la imagen de capa de referencia no está habilitado (rama "NO" de 252), el codificador de vídeo puede habilitar la predicción entre capas para la imagen de capa de mejora (258). El codificador de vídeo puede entonces codificar un bloque de la capa de mejora usando la predicción entre capas (260). En particular, después de haber codificado uno o más de los elementos de capa de referencia, el codificador de vídeo puede codificar bloques de un elemento de la imagen de capa de mejora que tiene límites alineados con el elemento de capa de referencia codificada. Además, el codificador de vídeo puede codificar los bloques del elemento de la imagen de capa de mejora en paralelo con un elemento posterior de la imagen de capa de referencia.

[0178] Debe entenderse que la imagen de capa de mejora y la imagen de capa de referencia están en general incluidas en la misma unidad de acceso. Es decir, la imagen de capa de mejora y la imagen de capa de referencia corresponden al mismo tiempo de salida (o tiempo de visualización).

[0179] De esta manera, el procedimiento de la FIG. 15 representa un ejemplo de un procedimiento que incluye, al codificar (por ejemplo, codificar o descodificar) un bloque de capa de mejora en un elemento de capa de mejora, evitar la codificación del bloque de capa de mejora usando predicción de textura entre capas o predicción de sintaxis entre capas a partir de un bloque de capa base colocado para el cual está habilitado el filtrado entre capas o el filtrado de capa de referencia en una imagen de capa de referencia en una unidad de acceso que incluye tanto el elemento de capa de mejora como el bloque de capa base, y codificar (codificar o descodificar) el bloque colocado sustancialmente en paralelo al bloque de la capa de mejora cuando se habilita el filtrado entre capas o el filtrado de

capa de referencia a lo ancho de los límites de los elementos en la imagen de capa de referencia. De forma alternativa, cuando el filtrado entre capas y el filtrado de capa de referencia a lo ancho de los límites de los elementos en la imagen de capa de referencia no están habilitados, el procedimiento puede incluir codificación (codificación o decodificación) del bloque de capa de mejora en paralelo con un elemento posterior de la imagen de capa de referencia.

[0180] La FIG. 16 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar imágenes de una capa de mejora y una capa de referencia en paralelo cuando las imágenes tienen relaciones de aspecto diferentes, de acuerdo con las técnicas de esta divulgación. El procedimiento de la FIG. 16 en general se describe como realizado por un codificador de vídeo. El codificador de vídeo puede corresponder, por ejemplo, al codificador de vídeo 20 o al decodificador de vídeo 30, u otros dispositivos codificadores de vídeo (por ejemplo, dispositivos de transcodificación de vídeo).

[0181] En este ejemplo, el codificador de vídeo primero codifica un elemento de una imagen de capa de referencia que tiene una primera relación de aspecto (280). Por ejemplo, la imagen de capa de referencia puede tener una relación de aspecto de 4:3. El codificador de vídeo puede entonces determinar si una imagen de capa de mejora tiene una relación de aspecto que sea mayor que la relación de aspecto de la imagen de capa de referencia (282). Por ejemplo, la imagen de capa de mejora puede tener una relación de aspecto de 16:9, mientras que la imagen de capa de referencia puede tener una relación de aspecto de 4:3.

[0182] Si la imagen de capa de mejora tiene una relación de aspecto que es mayor que la relación de aspecto de la imagen de capa de referencia (rama "SI" de 282), habrá al menos algunas partes (por ejemplo, elementos) de la imagen de capa de mejora que no se superpongan con la imagen de capa de referencia. Por lo tanto, el codificador de vídeo puede codificar partes superpuestas de la imagen de capa de mejora usando predicción entre capas (284), y codificar partes no superpuestas de la imagen de capa de mejora sin usar predicción entre capas (286). Además, el codificador de vídeo puede codificar partes no superpuestas de la imagen de capa de mejora en paralelo con cualquier parte de la imagen de capa de referencia, mientras que el codificador de vídeo puede codificar partes superpuestas de la imagen de capa de mejora en paralelo con partes posteriores de la imagen de capa de referencia. Por ejemplo, haciendo referencia de nuevo a la FIG. 13, el codificador de vídeo puede codificar los elementos 184A, 184D en paralelo con cualquier parte de la imagen de capa de referencia 182. El codificador de vídeo puede codificar el elemento 184B en paralelo con el elemento 186B.

[0183] Por otro lado, si la imagen de capa de mejora no tiene una relación de aspecto mayor que la relación de aspecto de la imagen de capa de referencia (rama "NO" de 282), el codificador de vídeo puede codificar completamente la imagen de capa de mejora utilizando predicción entre capas (288), y puede codificar ciertos elementos de la imagen de capa de mejora en paralelo con otros elementos de la imagen de capa de referencia.

[0184] De esta manera, el procedimiento de la FIG. 16 representa un ejemplo de un procedimiento que incluye codificación (por ejemplo, codificación o decodificación) de una imagen de capa base que comprende una primera relación de aspecto, y codificación (por ejemplo, codificación o decodificación) sustancialmente en paralelo con la imagen de capa base, de una imagen de capa de mejora que comprende una segunda relación de aspecto, en el que la segunda relación de aspecto es mayor que la primera relación de aspecto, y en el que la codificación de la imagen de capa de mejora comprende partes de codificación de la imagen de capa de mejora que están fuera de un área definida por la primera relación de aspecto sin usar predicción entre capas.

[0185] La FIG. 17 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar una imagen de capa de mejora basándose en si los límites de los elementos de la imagen de capa de mejora están alineados con los límites de los elementos de una imagen de capa de referencia. El procedimiento de la FIG. 17 en general se describe como realizado por un codificador de vídeo. El codificador de vídeo puede corresponder, por ejemplo, al codificador de vídeo 20 o al decodificador de vídeo 30, u otros dispositivos codificadores de vídeo (por ejemplo, dispositivos de transcodificación de vídeo).

[0186] En este ejemplo, el codificador de vídeo codifica uno o más elementos de capa de referencia (300). El codificador de vídeo entonces determina si los límites de los elementos de capa de mejora están alineados (302). Por ejemplo, el codificador de vídeo 20 puede determinar si alinear los límites de la capa de referencia y puede codificar un valor para un elemento sintáctico que indique si los límites de los elementos están alineados, como `tile_boundaries_aligned_flag`, mientras que el decodificador 30 puede determinar si los límites de la capa de referencia están alineados, por ejemplo basándose en un valor de un elemento sintáctico, como `tile_boundaries_aligned_flag`. Se puede decir que los límites de los elementos de la imagen de capa de mejora están alineados con los límites de los elementos de la imagen de capa de referencia cuando, para dos muestras cualesquiera de imagen de capa de mejora que se encuentren dentro del mismo elemento de caja de mejora, las muestras de capa de referencia colocadas, si están disponibles, también se encuentran dentro del mismo elemento de capa de referencia, y para dos muestras cualesquiera de imagen de capa de referencia que se encuentren dentro del mismo elemento de capa de referencia, las muestras de capa de mejora colocadas, si están disponibles, también se encuentran dentro del mismo elemento de capa de mejora.

[0187] Cuando los límites de los elementos de la imagen de capa de mejora no están alineados con los límites de los elementos de la imagen de capa de referencia (rama "NO" de 302), el codificador de vídeo puede inhabilitar la predicción entre capas para la capa de mejora (304) y codificar los elementos de la imagen de capa de mejora sin usar la predicción entre capas (306). Además, el codificador de vídeo puede codificar cualquiera o todos los elementos de la imagen de capa de mejora en paralelo con cualquier parte de la imagen de capa de referencia, porque la predicción entre capas está inhabilitada.

[0188] Por otro lado, cuando los límites de los elementos de la imagen de capa de mejora están alineados con los límites de los elementos de la imagen de capa de referencia (rama "SÍ" de 302), el codificador de vídeo puede habilitar la predicción entre capas para la imagen de capa de mejora (308) y codificar los elementos de la imagen de capa de mejora usando predicción entre capas (310). En particular, después de codificar un elemento de la imagen de capa de referencia, el codificador de vídeo puede comenzar a codificar un elemento correspondiente de la imagen de capa de mejora en paralelo con un elemento posterior de la imagen de capa de referencia.

[0189] Si hay elementos de la imagen de capa de mejora que no se corresponden con la imagen de capa de referencia (por ejemplo, como se analizó anteriormente con respecto a la FIG. 16), el codificador de vídeo puede codificar estos elementos en paralelo con cualquier parte de la imagen de capa de referencia. Incluso cuando tales elementos existen, aún se puede decir que los límites de los elementos están alineados, porque las muestras "colocadas" para tales elementos en la imagen de capa de referencia no estarán disponibles, según la definición de ejemplo proporcionada anteriormente.

[0190] De esta manera, el procedimiento de la FIG. 17 representa un ejemplo de un procedimiento que incluye codificar (por ejemplo, codificar o decodificar) un valor para un elemento sintáctico representativo de si dos muestras cualesquiera de capa de referencia, colocadas con dos muestras de imagen de capa de mejora respectivas dentro de un elemento de capa de mejora común, deben estar dentro de una elemento de capa de referencia común, y codificar (por ejemplo, codificar o decodificar) las muestras de imagen de capa de mejora basándose, al menos en parte, en el valor del elemento sintáctico.

[0191] La FIG. 18 es un diagrama de flujo que ilustra un procedimiento de ejemplo para codificar elementos de una imagen de capa de mejora en paralelo con una imagen de capa de referencia, incluso cuando los límites de los elementos no están alineados, de acuerdo con las técnicas de esta divulgación. El procedimiento de la FIG. 18 en general se describe como realizado por un codificador de vídeo. El codificador de vídeo puede corresponder, por ejemplo, al codificador de vídeo 20 o al decodificador de vídeo 30, u otros dispositivos codificadores de vídeo (por ejemplo, dispositivos de transcodificación de vídeo).

[0192] El codificador de vídeo puede determinar un número de elementos de capa de referencia (es decir, un cierto número de elementos en una imagen de capa de referencia) que tienen que codificarse antes de codificar un elemento de capa de mejora particular (es decir, un elemento de una imagen de capa de mejora correspondiente) (320). En particular, el codificador de vídeo puede determinar un número de elementos de capa de referencia a codificar para cada elemento de capa de mejora. Por ejemplo, suponiendo que la codificación avanza en un orden de escaneado de trama (de izquierda a derecha, de arriba a abajo), el codificador de vídeo 20 puede determinar un número de elementos de capa de referencia que deben codificarse antes de que pueda codificarse un elemento de capa de mejora particular, es decir, antes de que se hayan codificado todas las muestras colocadas en el elemento de capa de referencia para el elemento de capa de mejora. Aunque el orden de escaneado de trama se utiliza para fines de ejemplo, se pueden usar otros órdenes de escaneado de elemento. El codificador de vídeo 20 puede codificar un valor para un elemento sintáctico correspondiente al elemento de capa de mejora, por ejemplo, un valor para $\text{max_ref_tile_dec_idc_minus1}[i]$, donde i representa un índice para el elemento de capa de mejora en el orden de escaneado de trama. El decodificador de vídeo 30 puede decodificar un valor para el elemento sintáctico para determinar un número de elementos a decodificar antes de decodificar el elemento de capa de mejora. Los datos que indican el número de elementos de capa de referencia que se tienen que codificar pueden formar parte de un conjunto de parámetros de vídeo (VPS), un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (SPS), una cabecera de fragmento, información de usabilidad de vídeo (VUI) y/o un mensaje de información de mejora complementaria (SEI).

[0193] El codificador de vídeo puede entonces determinar si el número de elementos de capa de referencia se ha codificado (322). De no ser así, (rama "NO" de 322), el codificador de vídeo puede codificar un elemento de capa de referencia siguiente (326). El codificador de vídeo puede proceder a codificar elementos de capa de referencia hasta que se haya codificado el número determinado de elementos de capa de referencia. Después de que se haya codificado el número de elementos de capa de referencia (rama "SÍ" de 322), el codificador de vídeo puede codificar el elemento de capa de mejora en paralelo con uno o más elementos de capa de referencia (324).

[0194] De esta manera, el procedimiento de la FIG. 18 representa un ejemplo de un procedimiento que incluye codificar (por ejemplo, codificar o decodificar), para un elemento de una imagen de capa de mejora, datos que indican un número de elementos en una imagen de capa base que deben codificarse antes de que el elemento de la imagen de capa de mejora se pueda codificar, codificar (por ejemplo, codificar o decodificar) elementos de la imagen de capa base correspondiente al número de elementos y, después de codificar los elementos de la imagen

de capa base, codificar (por ejemplo, codificar o decodificar) el elemento de la imagen de capa de mejora sustancialmente en paralelo con al menos otro elemento en la imagen de capa base.

[0195] Debe reconocerse que, dependiendo del ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden agregarse, fusionarse u omitirse por completo (por ejemplo, no todos los actos o sucesos descritos son necesarios para la puesta en práctica de las técnicas). Además, en ciertos ejemplos, los actos o sucesos pueden realizarse simultáneamente, por ejemplo, mediante el procesamiento de múltiples hebras, el procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente.

[0196] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones, como una o más instrucciones o código, pueden almacenarse en, y transmitirse por, un medio legible por ordenador, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como 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.

[0197] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender memoria RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda utilizarse para almacenar un código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe adecuadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una sede de la Red, un servidor u otro origen remoto 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 reproducen datos magnéticamente, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior deberían incluirse también dentro del alcance de los medios legibles por ordenador.

[0198] 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 de aplicación específica (ASIC), matrices de puertas programables in situ (FPGA) u otra circuitería lógica equivalente, integrada o discreta. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

[0199] Las técnicas de la 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.

REIVINDICACIONES

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

5 descodificar una imagen de capa base que tiene una primera relación de aspecto;

10 descodificar datos representativos de si un elemento de una imagen de capa de mejora, que corresponde a la imagen de capa base, puede predecirse usando predicción entre capas, en el que el elemento comprende un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneado de trama de bloques de árbol de codificación del elemento, en el que la imagen de capa de mejora tiene una segunda relación de aspecto, en el que la segunda relación de aspecto es mayor que la primera relación de aspecto y en el que cuando el elemento de la imagen de capa de mejora está fuera de una parte de la imagen de capa de mejora que corresponde a la imagen de capa base, descodificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprende descodificar datos que indican que el elemento no puede predecirse usando la predicción entre capas;

15 predecir datos del elemento usando la predicción entre capas solo cuando los datos indican que el elemento puede predecirse usando predicción entre capas; y

20 descodificar, en paralelo con la imagen de capa base, la imagen de capa de mejora.

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

25 codificar una imagen de capa base que comprende una primera relación de aspecto;

30 codificar datos representativos de si un elemento de una imagen de capa de mejora, que corresponde a la imagen de capa base, puede predecirse usando predicción entre capas, en el que el elemento comprende un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneado de trama de bloques de árbol de codificación del elemento, en el que la imagen de capa de mejora tiene una segunda relación de aspecto, en el que la segunda relación de aspecto es mayor que la primera relación de aspecto y en el que cuando el elemento de la imagen de capa de mejora está fuera de una parte de la imagen de capa de mejora que corresponde a la imagen de capa base, codificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprende codificar datos que indican que el elemento no puede predecirse usando predicción entre capas;

35 predecir datos del elemento usando la predicción entre capas solo cuando los datos indican que el elemento puede predecirse usando predicción entre capas; y

40 codificar, en paralelo con la imagen de capa base, la imagen de capa de mejora.

3. El procedimiento según la reivindicación 1 o 2, que comprende además descodificar o codificar datos que indican que los elementos de la imagen de capa de mejora están alineados con los elementos de la imagen de capa base cuando el elemento de capa de mejora está fuera de la parte de la imagen de capa de mejora que corresponde a la imagen de capa base.

4. El procedimiento según la reivindicación 1 o 2, en el que descodificar o codificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprende datos de descodificación que especifican un conjunto de parámetros de imagen, PPS y datos que indican si puede usarse predicción entre capas de textura o sintaxis para los elementos de la imagen de capa de mejora.

5. El procedimiento según la reivindicación 4, en el que los datos que especifican el PPS comprenden un elemento sintáctico sei_pic_parameter_set_id.

6. El procedimiento según la reivindicación 1 o 2, en el que los datos que indican si se puede usar predicción entre capas de textura o sintaxis comprenden un non_interlayer_pred_flag que corresponde al elemento.

7. El procedimiento según la reivindicación 1 o 2, en el que descodificar los datos representativos de si el elemento puede predecirse usando la predicción entre capas comprende descodificar un mensaje SEI tile_interlayer_pred_info.

8. Un dispositivo para descodificar datos de vídeo, comprendiendo el dispositivo:

65 medios para descodificar una imagen de capa base que tiene una primera relación de aspecto;

medios para descodificar datos representativos de si un elemento de una imagen de capa de mejora puede predecirse usando predicción entre capas, en el que el elemento comprende un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneado de trama de bloques de árbol de codificación del elemento, en el que la imagen de capa de mejora tiene una segunda relación de aspecto, en el que la segunda relación de aspecto es mayor que la primera relación de aspecto, y en el que los medios para codificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprenden medios para codificar datos que indican que el elemento no puede predecirse usando predicción entre capas cuando el elemento de la imagen de capa de mejora se encuentra fuera de una parte de la imagen de capa de mejora que corresponde a la imagen de capa base;

medios para predecir datos del elemento usando predicción entre capas solo cuando los datos indican que el elemento puede predecirse usando predicción entre capas; y

medios para descodificar, en paralelo con la imagen de capa base, la imagen de capa de mejora.

9. Un dispositivo para codificar datos de vídeo, comprendiendo el dispositivo:

medios para codificar una imagen de capa base que tiene una primera relación de aspecto;

medios para codificar datos representativos de si un elemento de una imagen de capa de mejora puede predecirse usando predicción entre capas, en el que el elemento comprende un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneado de trama de bloques de árbol de codificación del elemento, en el que la imagen de capa de mejora tiene una segunda relación de aspecto, en el que la segunda relación de aspecto es mayor que la primera relación de aspecto, y en el que los medios para codificar los datos representativos de si el elemento pueden predecirse usando predicción entre capas comprenden medios para codificar datos que indican que el elemento no puede predecirse usando la predicción entre capas cuando el elemento de la imagen de capa de mejora se encuentra fuera de una parte de la imagen de capa de mejora que corresponde a la imagen de capa base;

medios para predecir datos del elemento usando predicción entre capas solo cuando los datos indican que el elemento puede predecirse usando predicción entre capas; y

medios para codificar, en paralelo con la imagen de capa base, la imagen de capa de mejora.

10. El dispositivo según la reivindicación 8 o 9, que comprende además medios para descodificar o codificar datos que indican que los elementos de la imagen de capa de mejora están alineados con elementos de la imagen de capa base cuando el elemento de capa de mejora está fuera de la parte de la imagen de capa de mejora que corresponde a la imagen de capa base.

11. El dispositivo según la reivindicación 8 o 9, en el que los medios para descodificar o codificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprenden medios para codificar datos que especifican un conjunto de parámetros de imagen, PPS y datos que indican si se puede usar la predicción entre capas de textura o sintaxis para elementos de la imagen de capa de mejora.

12. El dispositivo de la reivindicación 11, en el que los datos que especifican el PPS comprenden un elemento sintáctico sei_pic_parameter_set_id.

13. El dispositivo según la reivindicación 8 o 9, en el que los datos que indican si se puede usar la predicción entre capas de textura o sintaxis comprenden un non_interlayer_pred_flag que corresponde al elemento.

14. El dispositivo según la reivindicación 8 o 9, en el que los medios para descodificar o codificar los datos representativos de si el elemento puede predecirse usando predicción entre capas comprende medios para descodificar o codificar un mensaje SEI tile_interlayer_pred_info.

15. Un medio legible por ordenador que tiene instrucciones almacenadas en el mismo que, cuando son ejecutadas, hacen que un procesador lleve a cabo el procedimiento según cualquiera de las reivindicaciones 1 a 7.

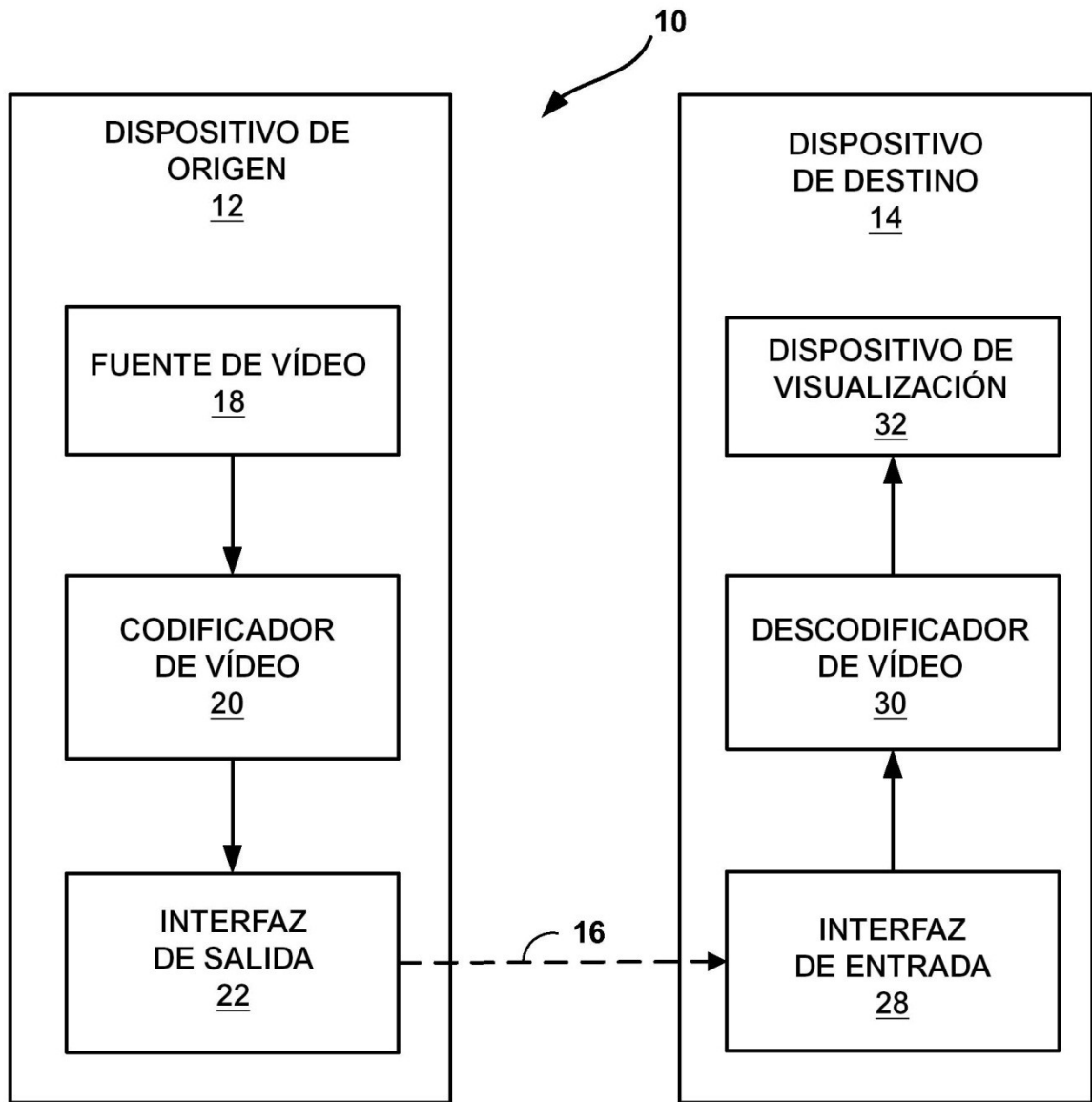


FIG. 1

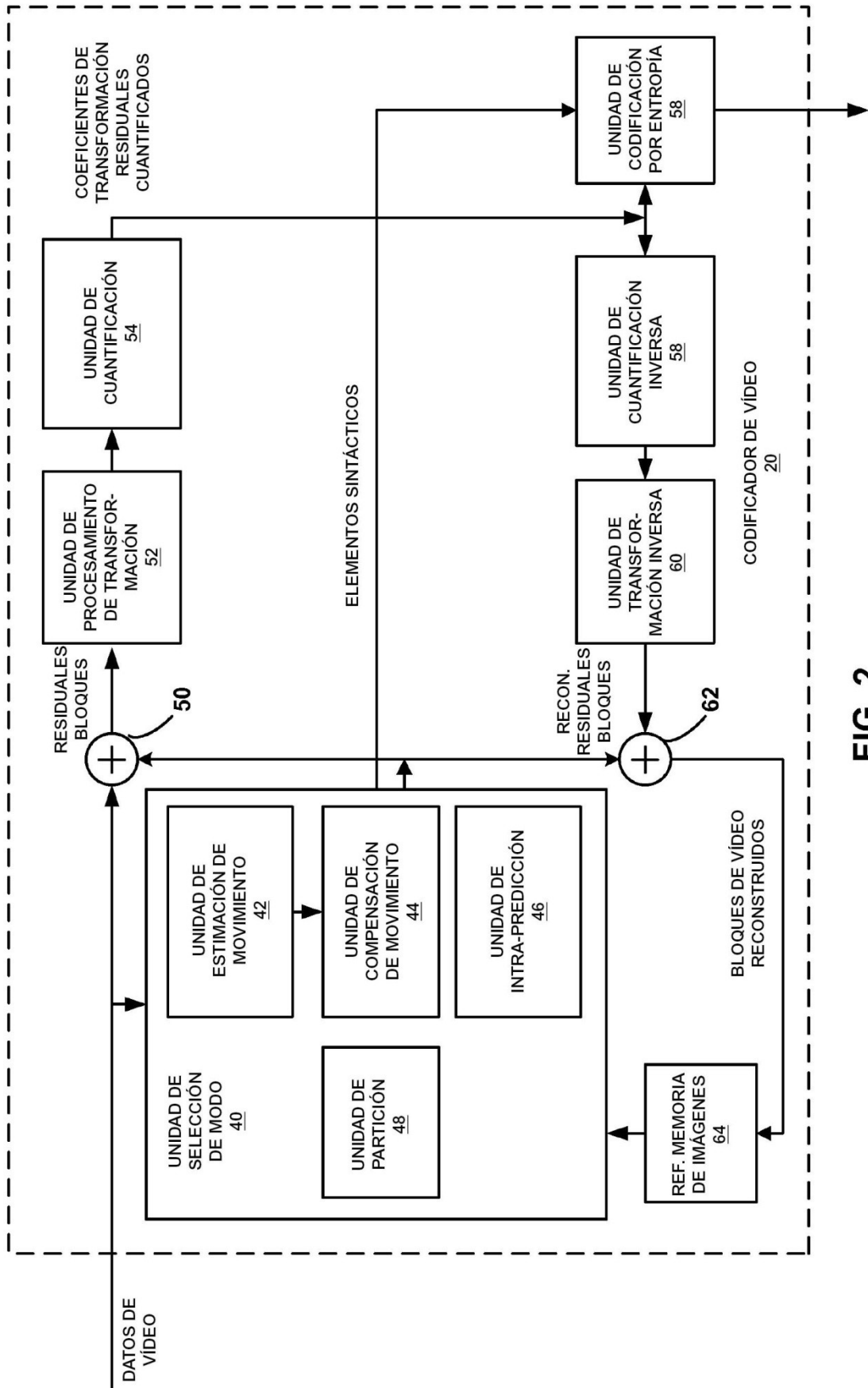


FIG. 2

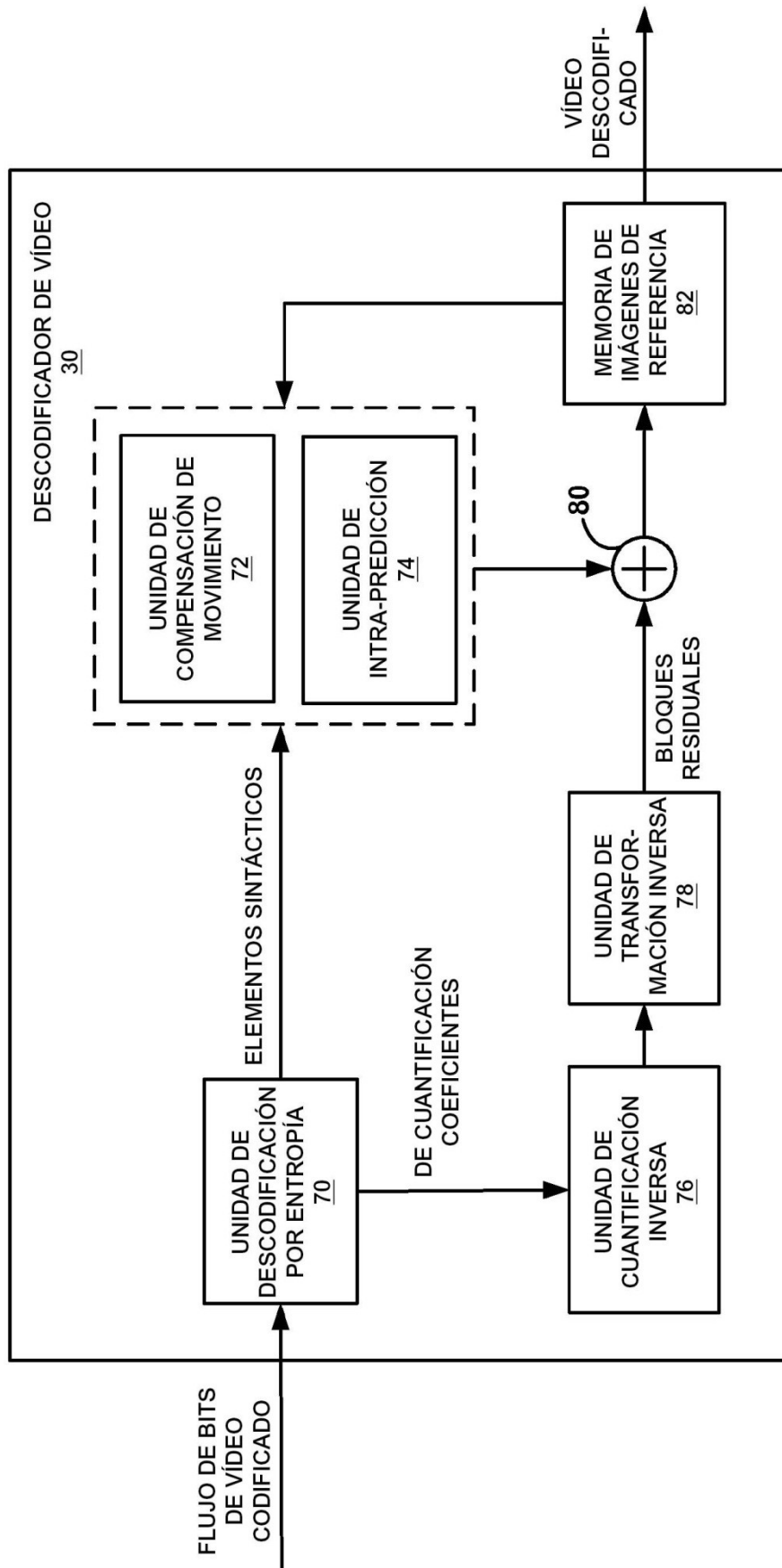


FIG. 3

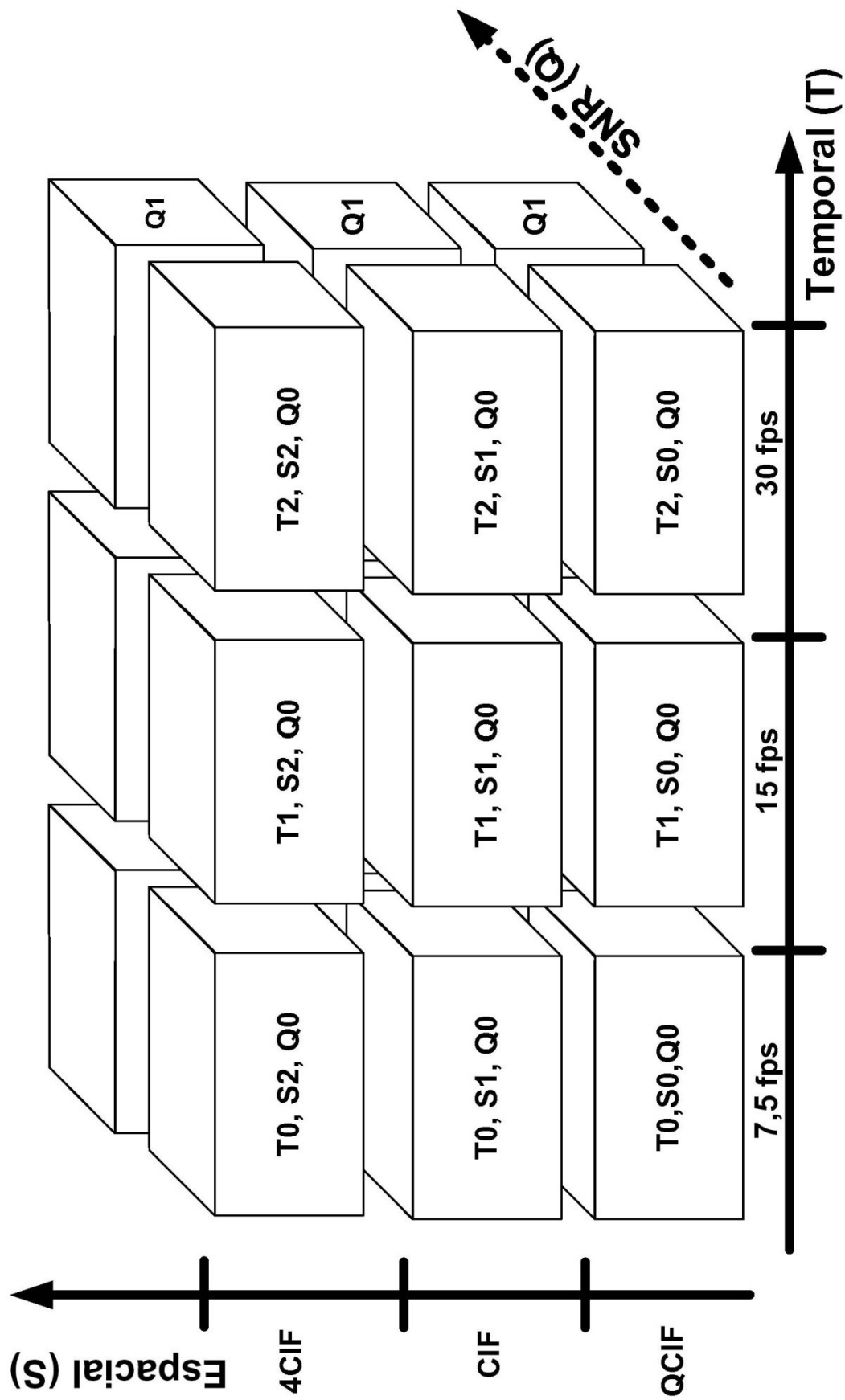


FIG. 4

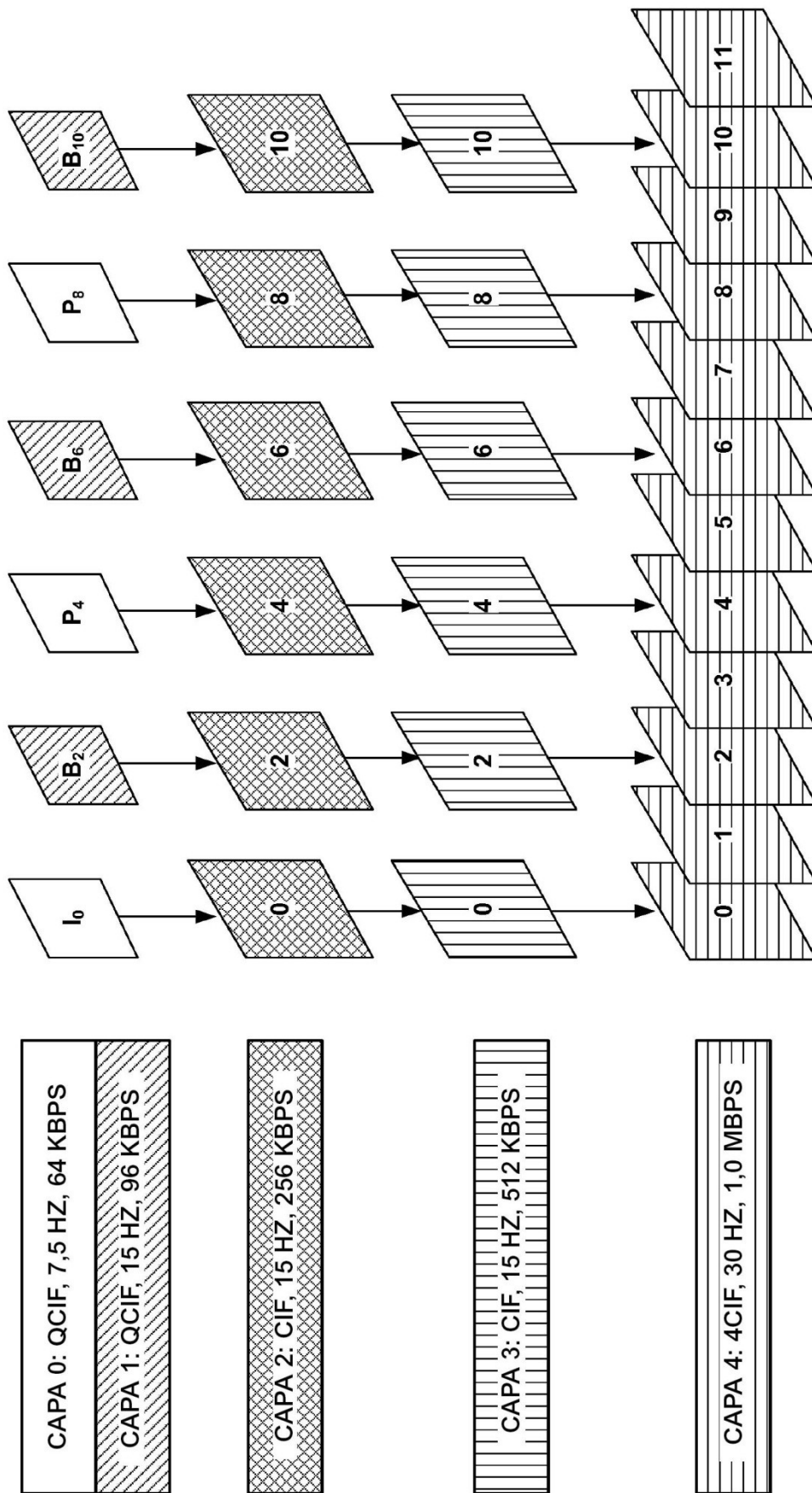


FIG. 5

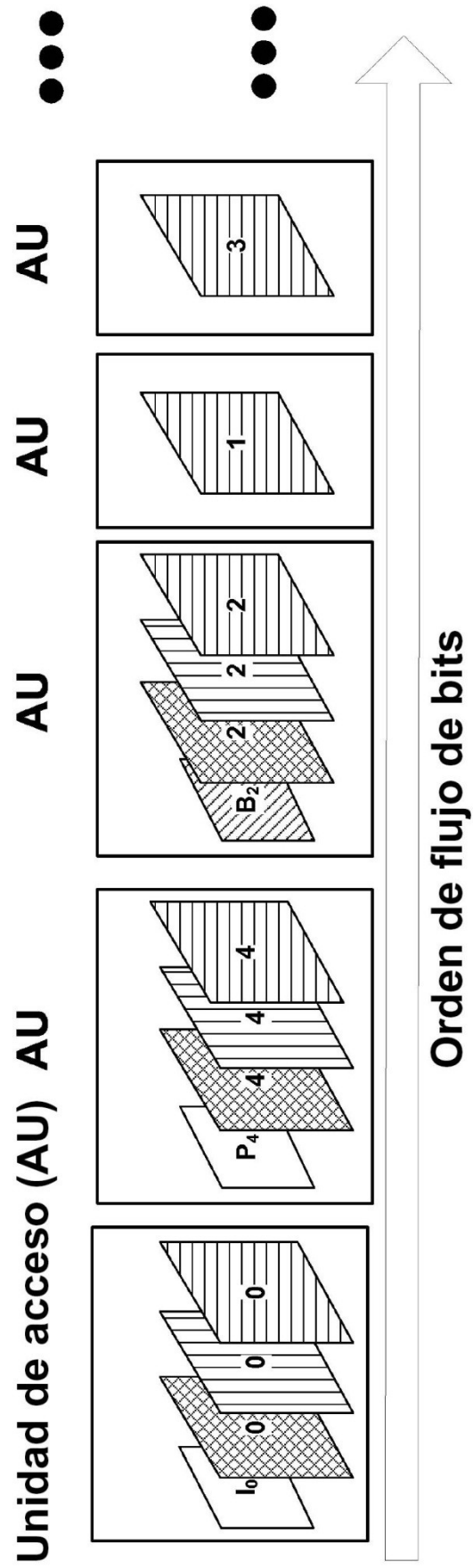


FIG. 6

0	1	2	3	12	13	14	21	22	23	24
4	5	6	7	15	16	17	25	26	27	28
8	9	10	11	18	19	20	29	30	31	32
33	34	35	36	41	42	43	47	48	49	50
37	38	39	40	44	45	46	51	52	53	54
55	56	57	58	63	64	65	69	70	71	72
59	60	61	62	66	67	68	73	74	75	76

FIG. 7

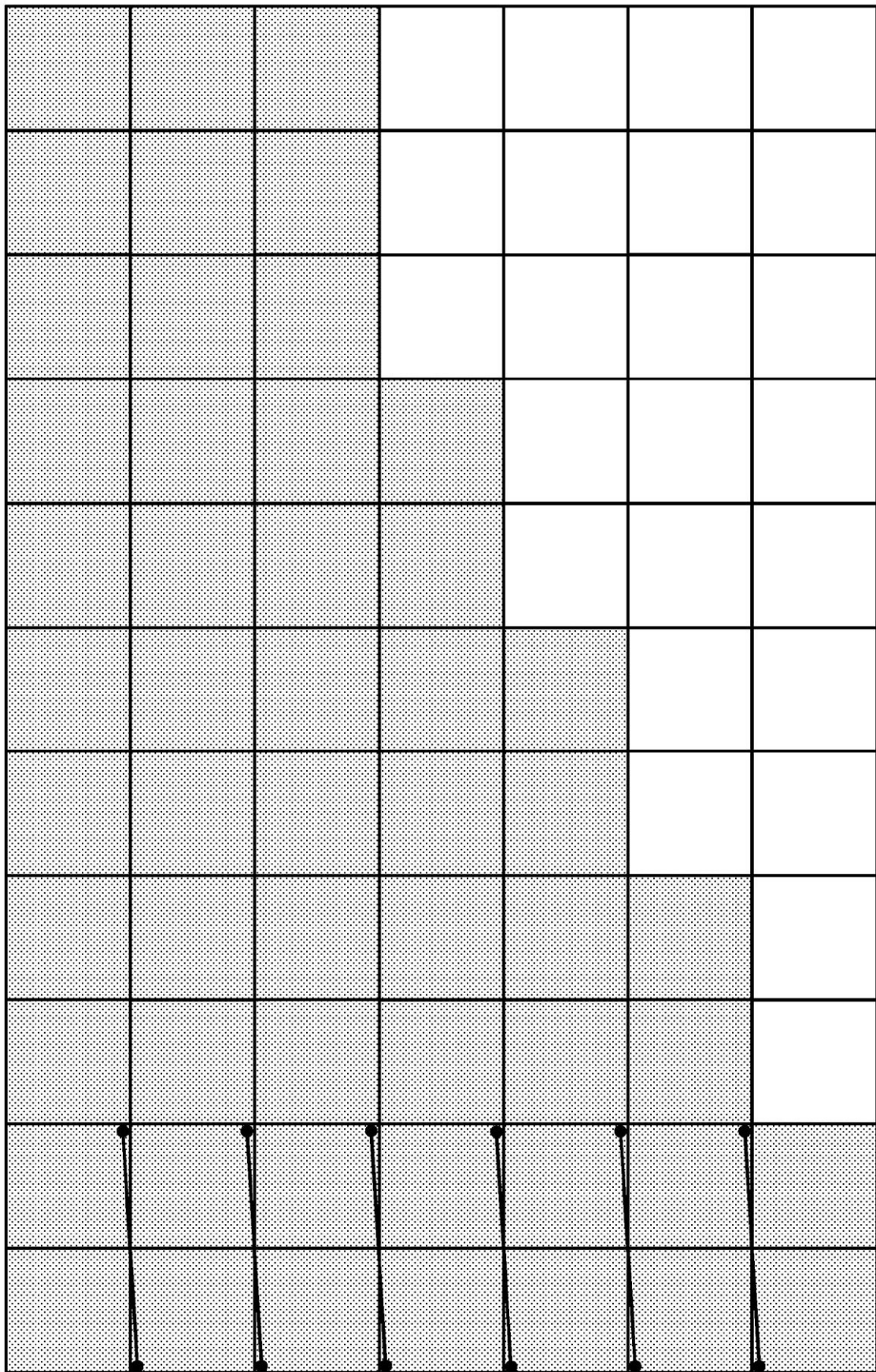


FIG. 8

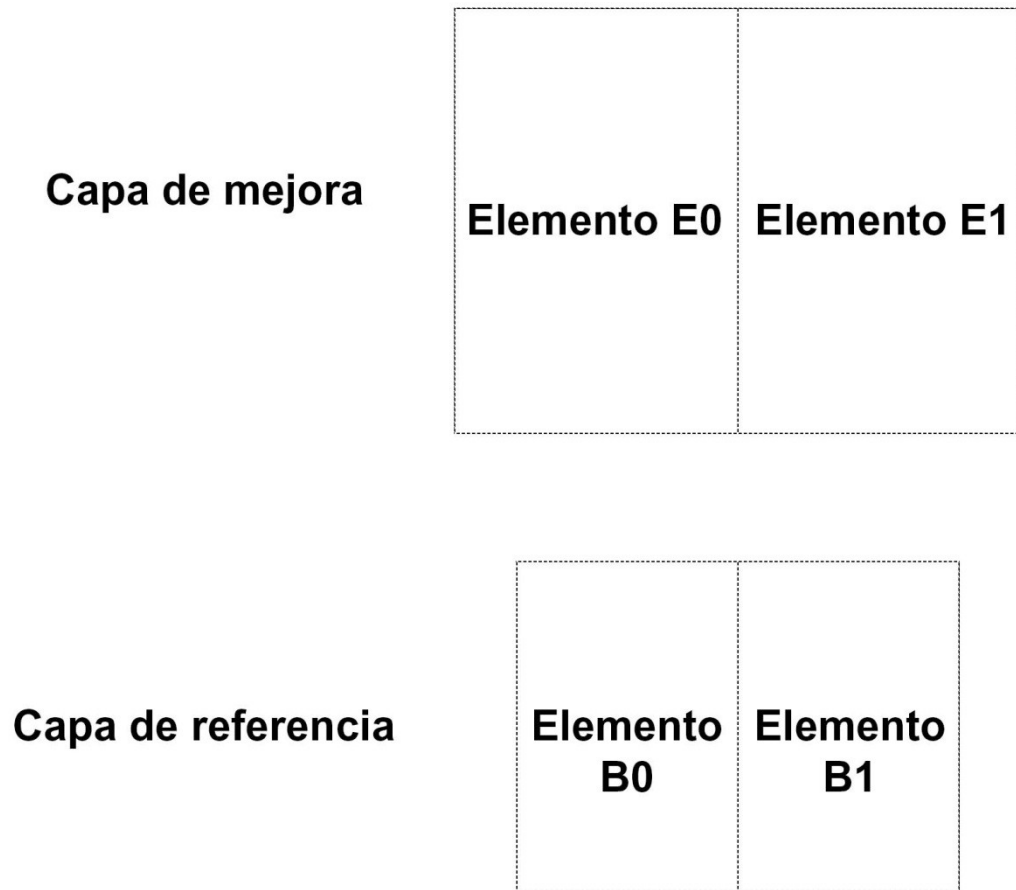
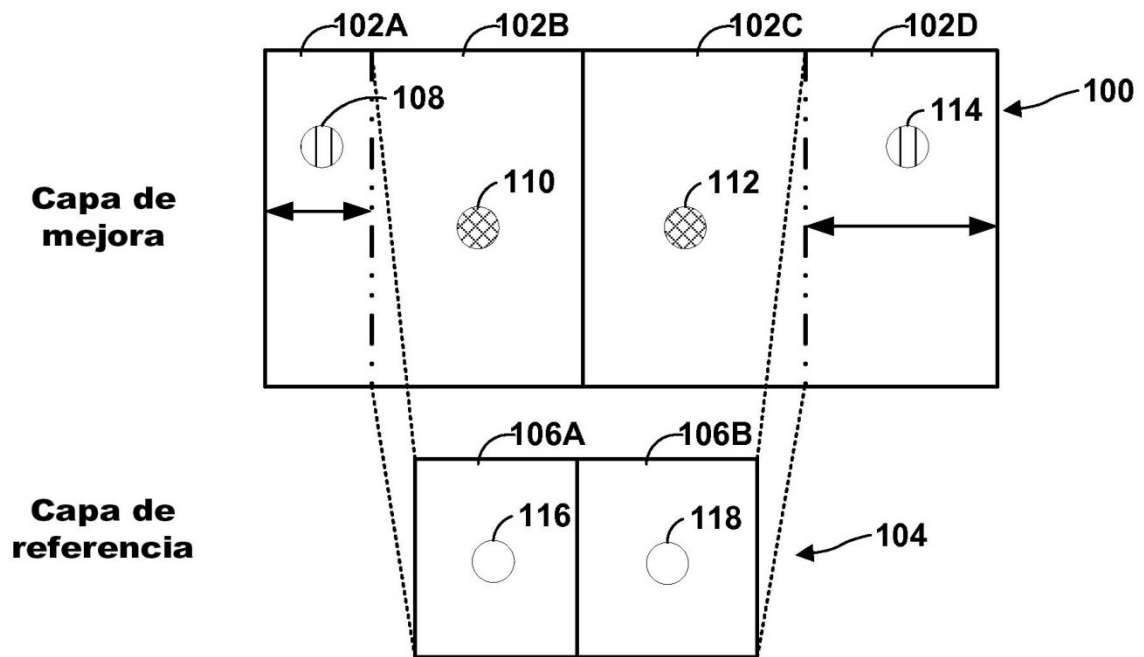


FIG. 9



 **Muestra de RL
colocada disponible**

 **Muestra de RL
colocada no disponible**

FIG. 10

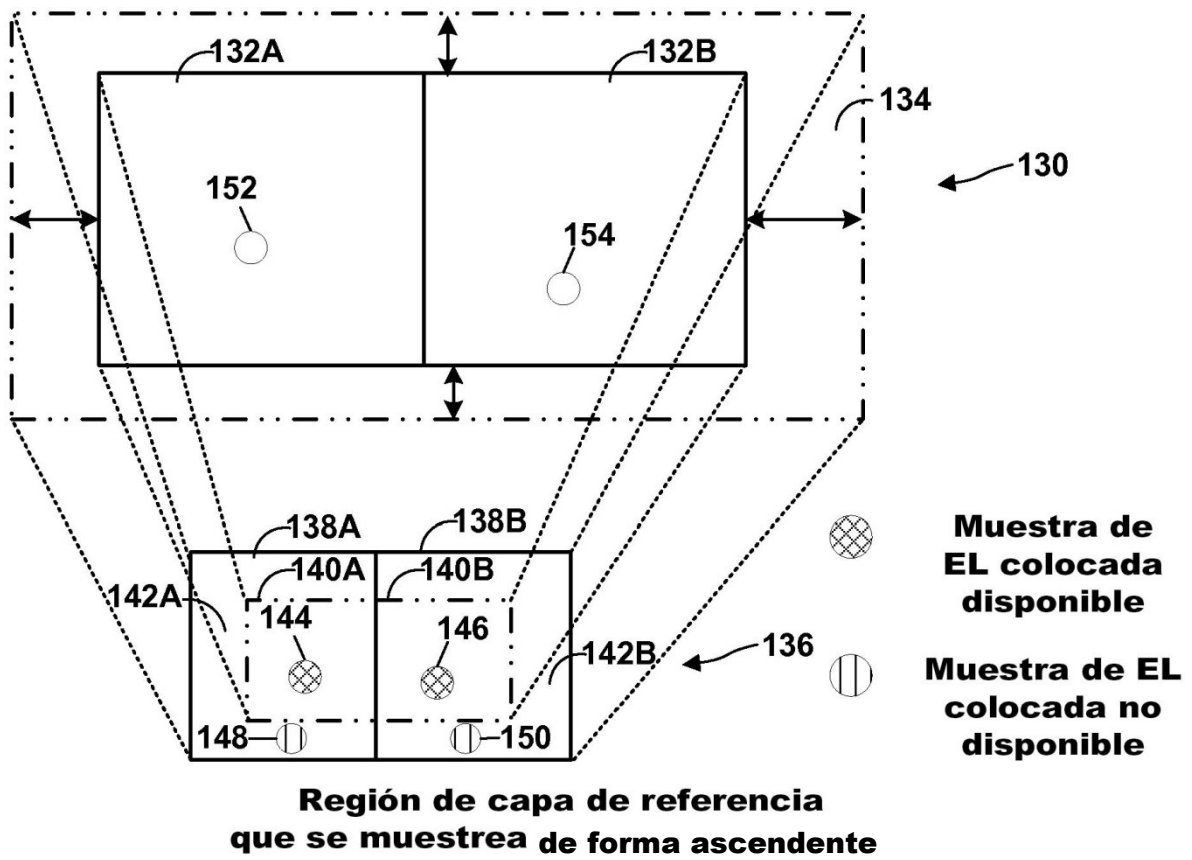


FIG. 11

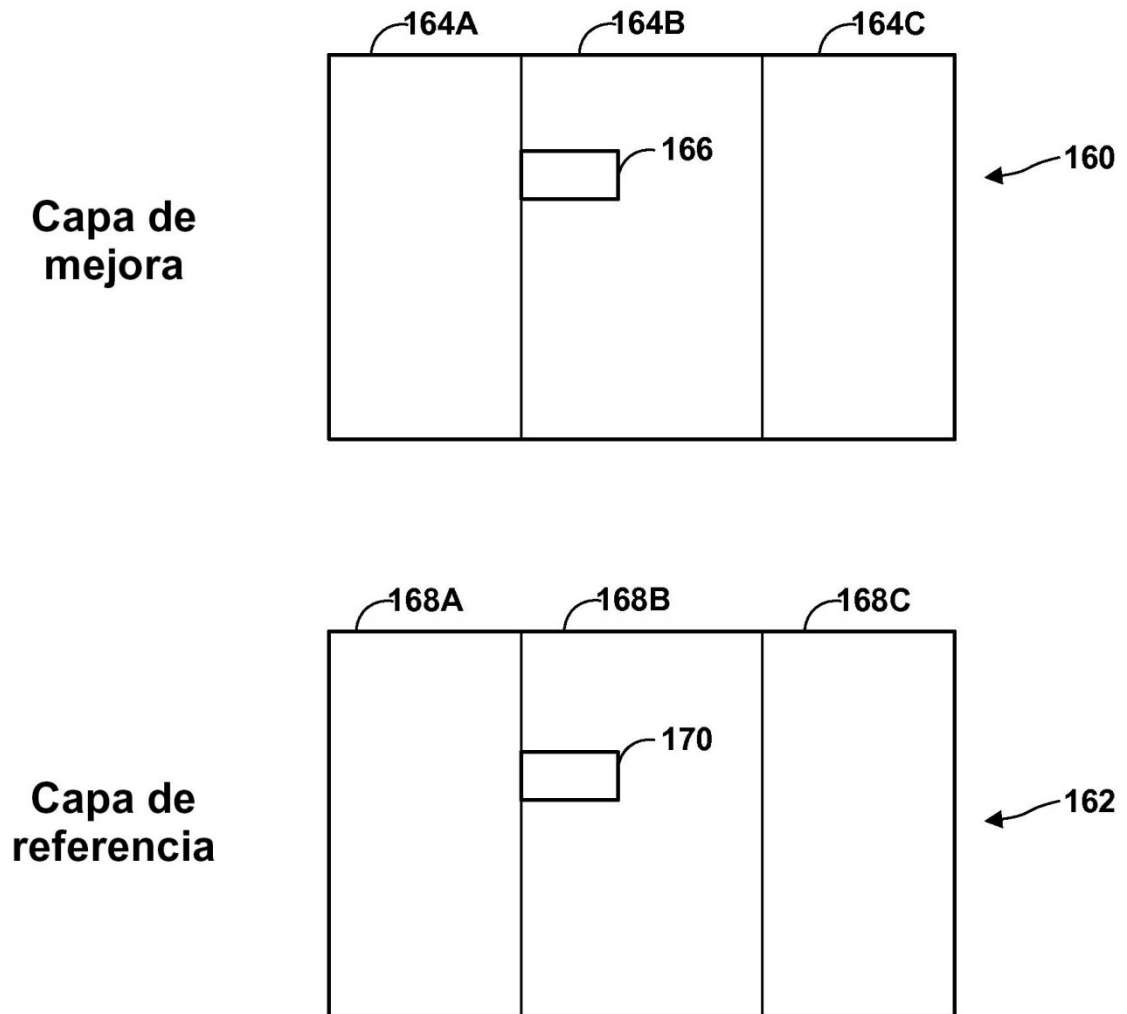


FIG. 12

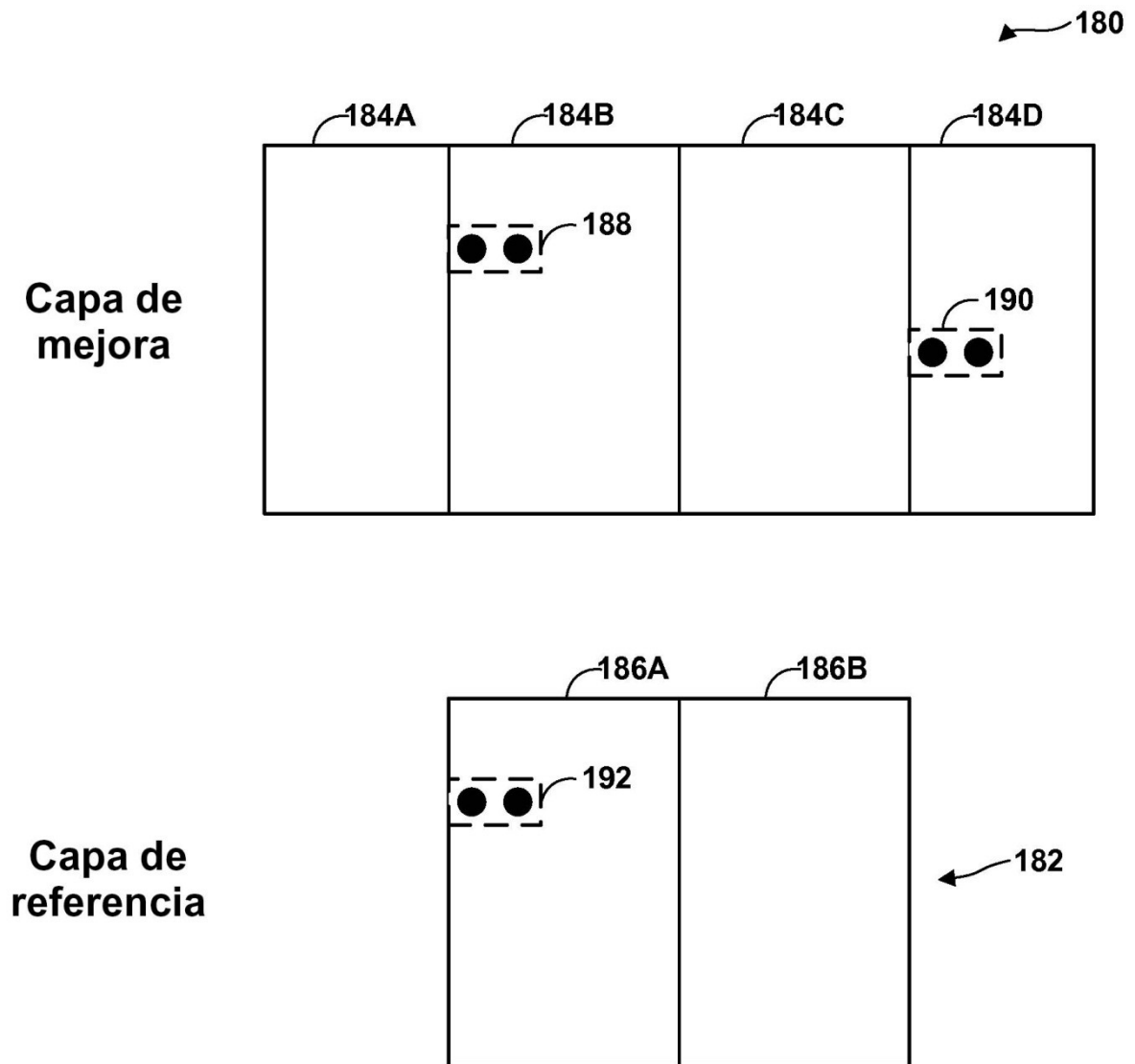


FIG. 13

**Capa de
mejora**

<u>204A</u>	<u>204B</u>	<u>204C</u>	<u>204D</u>
<u>204E</u>	<u>204F</u>	<u>204G</u>	<u>204H</u>

← 200

**Capa de
referencia**

<u>206A</u>	<u>206B</u>	<u>206C</u>
<u>206D</u>	<u>206E</u>	<u>206F</u>

← 202

FIG. 14

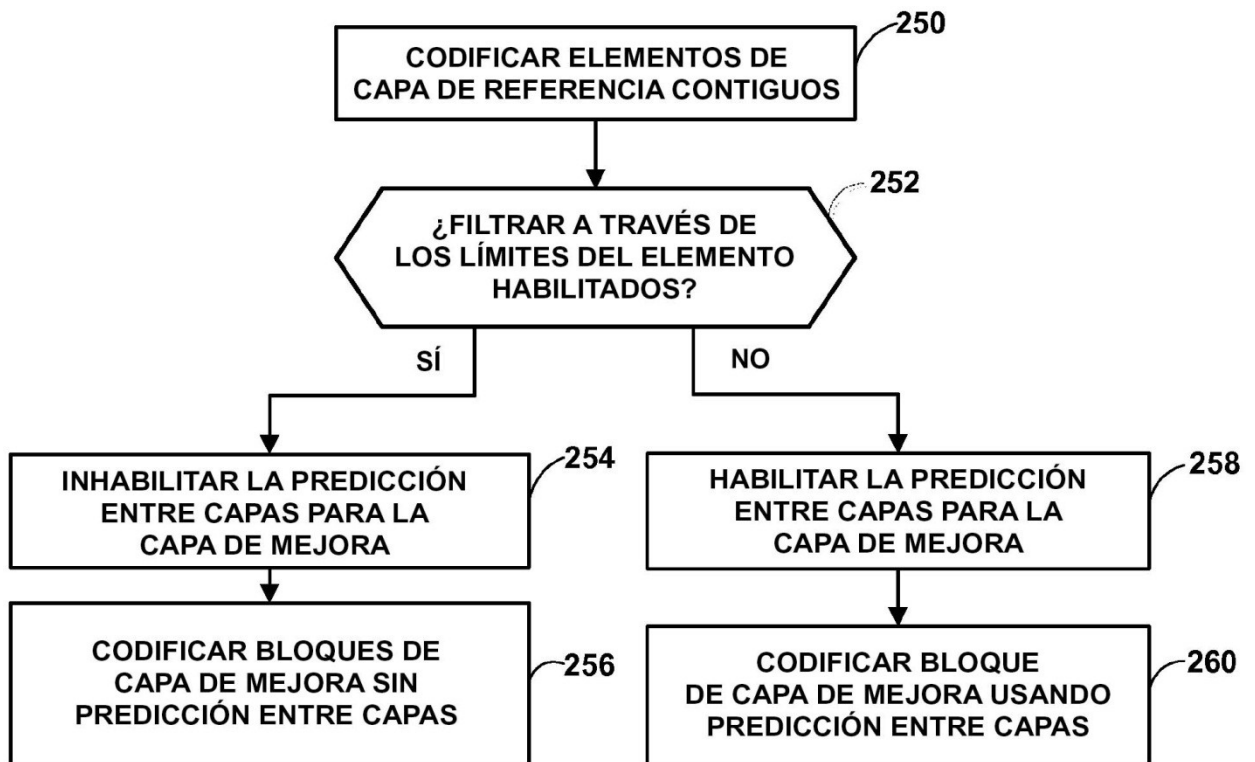


FIG. 15

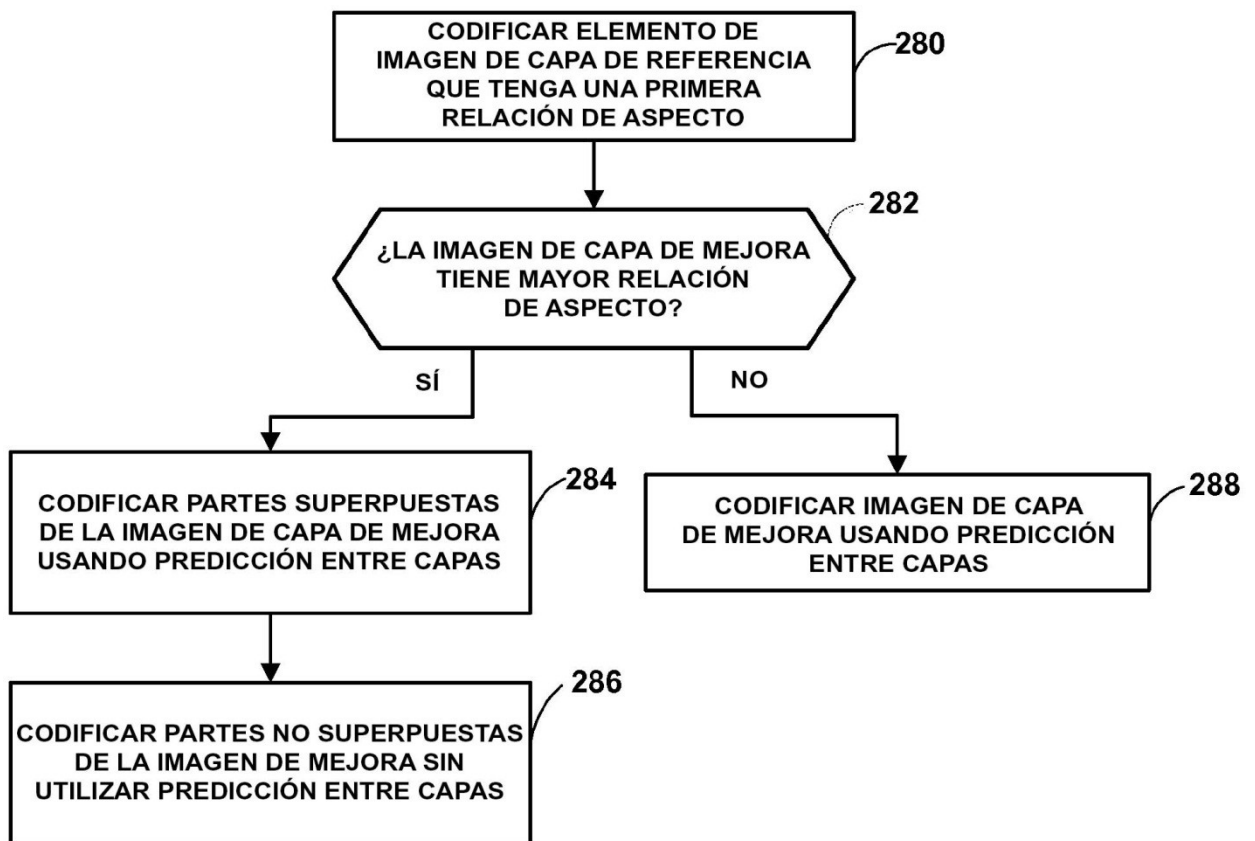


FIG. 16

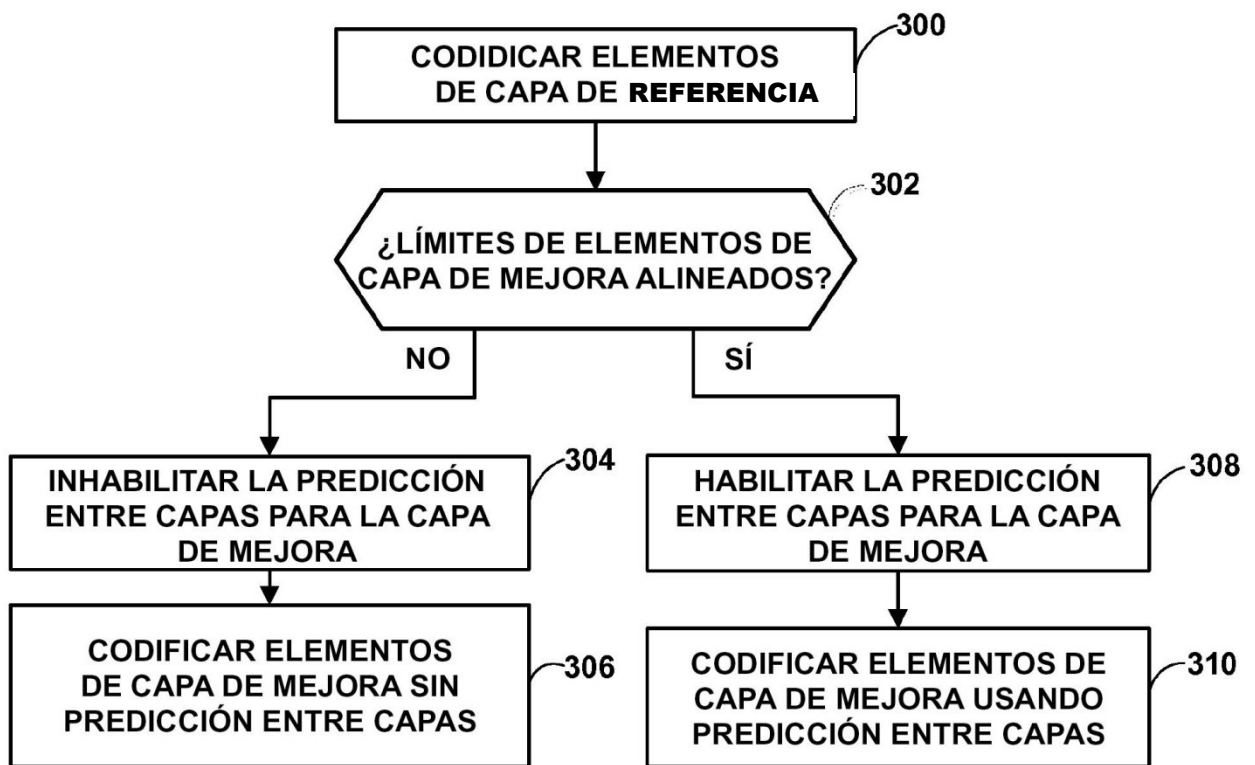


FIG. 17

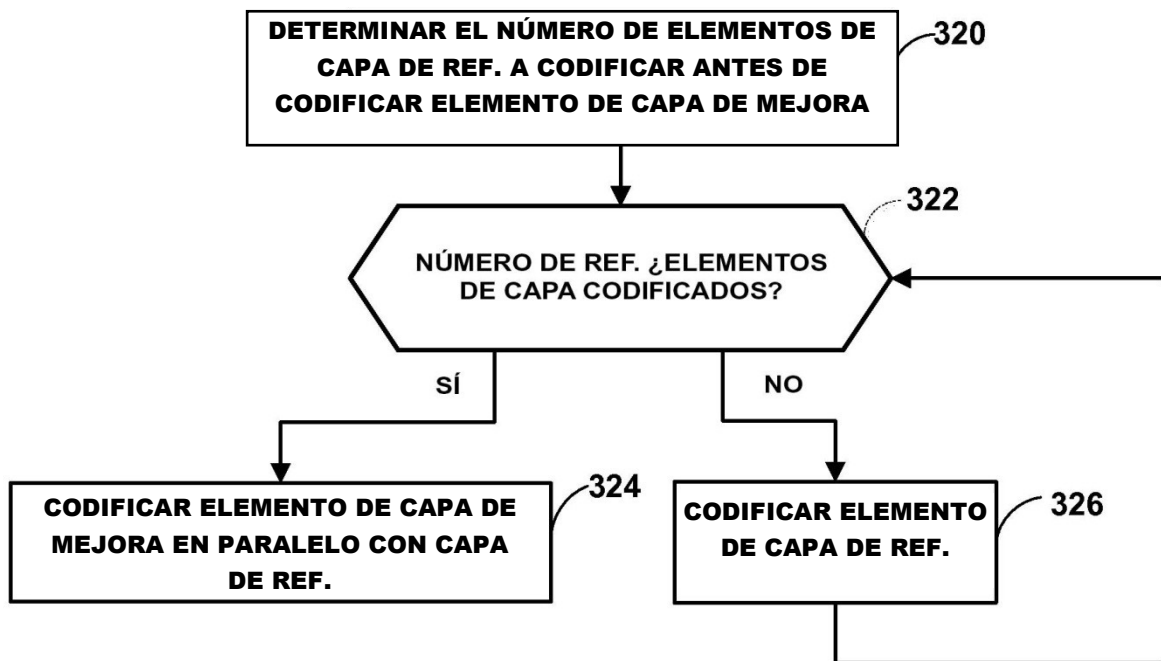


FIG. 18