

**ADAPTACIÓN DE IMÁGENES DE REFERENCIA A NIVEL DE BLOQUE
PARA CODIFICACIÓN DE VIDEO**

Esta solicitud reivindica prioridad a la solicitud
5 de patente provisional de los EE. UU. N.º 63/266.389,
presentada el 4 de enero de 2022, la solicitud de patente
provisional de los EE. UU. N.º 63/316.102, presentada el 3
de marzo de 2022, la solicitud de patente provisional de los
EE. UU. N.º 63/324.926, presentada el 29 de marzo de 2022,
10 la solicitud de patente provisional de los EE. UU. N.º
63/343.980, presentada el 19 de mayo de 2022, y la solicitud
de patente de los EE. UU. 18/068.809, presentada el 20 de
diciembre de 2022. La solicitud de patente de los EE. UU.
18/068.809 reivindica el beneficio de la solicitud de
15 patente provisional de los EE. UU. 63/266.389, la solicitud
de patente provisional de los EE. UU. 63/316.102, la
solicitud de patente provisional de los EE. UU. 63/324.926
y la solicitud de patente provisional de los EE. UU.
63/343.980, en donde el contenido de cada una de estas se
20 incorpora en la presente como referencia.

CAMPO TÉCNICO

Esta divulgación se relaciona con la
codificación de video y la decodificación de video.

25

ANTECEDENTES

Se pueden incorporar capacidades de video digital en una amplia gama de dispositivos, incluidos televisores digitales, sistemas de transmisión directa digital, sistemas de transmisión inalámbrica, asistentes digitales personales (PDA, por sus siglas en inglés), computadores portátiles o de escritorio, tabletas, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos de radio celulares o satelitales, los llamados “teléfonos inteligentes”, dispositivos de teleconferencia de video, dispositivos de transmisión de video, y similares. Los dispositivos de video digital implementan técnicas de codificación de video, tales como las descritas en los estándares definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación de video avanzada (AVC, por sus siglas en inglés), ITU-T H.265/Codificación de video de alta eficiencia (HEVC, por sus siglas en inglés), ITU-T H.266/Codificación de video versátil (VVC, por sus siglas en inglés), y extensiones de tales estándares, así como códecs/formatos de video de propiedad exclusiva, tales como AOMedia Video 1 (AV1, por sus siglas en inglés), que fue desarrollado por la Alliance for Open Media. Los dispositivos de video pueden transmitir, recibir, codificar, decodificar y/o almacenar

información de video digital de manera más eficiente implementando tales técnicas de codificación de video.

Las técnicas de codificación de video incluyen predicción espacial (intraimagen) y/o predicción temporal (interimagen) para reducir o eliminar la redundancia inherente de las secuencias de video. Para la codificación de video basada en bloques, un corte de video (por ejemplo, una imagen de video o una parte de una imagen de video) puede dividirse en bloques de video, que también pueden denominarse unidades de árbol de codificación (CTU, por sus siglas en inglés), unidades de codificación (CU, por sus siglas en inglés) y/o nodos de codificación. Los bloques de video en un corte intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de video en un corte intercodificado (P o B) de una imagen pueden usar predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen o predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse cuadros, y las imágenes de referencia pueden denominarse cuadros de referencia.

SUMARIO

Las técnicas de esta divulgación están dirigidas a la interpredicción en códecs de video, y, más

específicamente, a la señalización de imágenes de referencia y la gestión de listas de imágenes de referencia para bloques intercodificados. Con las técnicas existentes, un decodificador de video usa señalización de alto nivel, como a nivel de imagen o de corte, para mantener listas de imágenes de referencia para la interpredicción. Sin embargo, estas técnicas no incluyen adaptaciones a nivel de bloque a las listas de imágenes de referencia, como el reordenamiento. Incluir señalización adicional para permitir dicha adaptación a nivel de bloque puede ser indeseable debido a que el aumento de la sobrecarga de señalización supera las ganancias de codificación logradas por la adaptación de la lista de imágenes de referencia. Esta divulgación describe técnicas para realizar la adaptación de listas de imágenes de referencia a nivel de bloque sin aumentar la sobrecarga de señalización. Específicamente, un decodificador de video puede configurarse para realizar un proceso de reordenamiento de imágenes de referencia adaptable para generar un nuevo ordenamiento de las imágenes de referencia en una lista de imágenes de referencia sin sobrecarga de señalización. Al reordenar las imágenes de referencia en la lista de imágenes de referencia usando un proceso de reordenamiento de lista de imágenes de referencia adaptativo, los bits necesarios para señalar qué imagen de referencia en la lista de imágenes de referencia se usará para la interpredicción pueden reducirse sin aumentar la distorsión.

De acuerdo con un ejemplo de la divulgación, un método de decodificación de datos de video incluye generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte basado en un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento; decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del segundo bloque.

De acuerdo con un ejemplo de la divulgación, un dispositivo para decodificar datos de video incluye una memoria configurada para almacenar datos de video y uno o más procesadores implementados en circuitos y configurados para generar un primer ordenamiento de imágenes de

referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento; decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del segundo bloque.

De acuerdo con un ejemplo de la divulgación, un medio de almacenamiento legible por computadora almacena instrucciones que, cuando son ejecutadas por uno o más procesadores, hacen que el uno o más procesadores generen un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las

imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del
5 corte en función de un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos
10 de los índices a imágenes de referencia diferentes que en el primer ordenamiento; decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y
15 emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del segundo bloque.

De acuerdo con un ejemplo de la divulgación, un aparato para decodificar datos de video incluye medios para
20 generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; medios para
25 decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; medios para generar un segundo

ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es
5 diferente del segundo ordenamiento y los medios para generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprenden medios para asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento; medios para decodificar el
10 segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y medios para emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del segundo bloque.

Los detalles de uno o más ejemplos se exponen en
15 las figuras adjuntas y en la siguiente descripción. Otras características, objetos y ventajas resultarán evidentes a partir de la descripción, los dibujos y las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

20 La Figura 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de video de ejemplo que puede realizar las técnicas de estadivulgación.

La Figura 2 muestra un ejemplo de una plantilla que puede usarse junto con las técnicas descritas en la
25 presente.

La Figura 3 es un diagrama de bloques que ilustra un codificador de video de ejemplo que puede realizar las técnicas de esta divulgación.

La Figura 4 es un diagrama de bloques que ilustra
5 un decodificador de video de ejemplo que puede realizar las técnicas de esta divulgación.

La Figura 5 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque actual de acuerdo con las técnicas de esta divulgación.

10 La Figura 6 es un diagrama de flujo que ilustra un proceso de ejemplo para decodificar un bloque actual de acuerdo con las técnicas de esta divulgación.

La Figura 7 es un diagrama de flujo que ilustra un proceso de ejemplo para decodificar un bloque actual de
15 acuerdo con las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

La codificación de video (por ejemplo, codificación de video y/o decodificación de video) implica
20 típicamente predecir un bloque de datos de video a partir de un bloque ya codificado de datos de video en la misma imagen (por ejemplo, intrapredicción) o un bloque ya codificado de datos de video en una imagen diferente (por ejemplo, interpredicción). En algunos casos, el
25 codificador de video también calcula los datos residuales al comparar el bloque de predicción con el bloque original.

Por lo tanto, los datos residuales representan una diferencia entre el bloque de predicción y el bloque original. Para reducir el número de bits necesarios para señalar los datos residuales, el codificador de video
5 transforma y cuantifica los datos residuales y señala los datos residuales transformados y cuantificados en el flujo de bits codificado. La compresión lograda por los procesos de transformada y cuantificación puede tener pérdidas, lo que significa que los procesos de transformada y
10 cuantificación pueden introducir distorsión en los datos de video decodificados.

Un decodificador de video decodifica y añade los datos residuales al bloque de predicción para producir un bloque de video reconstruido que coincide con el bloque de
15 video original más estrechamente que el bloque de predicción solo. Debido a la pérdida introducida por la transformación y cuantificación de los datos residuales, el primer bloque reconstruido puede tener distorsión o artefactos. Un tipo común de artefacto o distorsión se conoce como “*blockiness*”
20 (fragmentación en bloques), donde los límites de los bloques usados para codificar los datos de video son visibles.

Para mejorar aún más la calidad del video decodificado, un decodificador de video puede realizar una o más operaciones de filtrado en los bloques de video
25 reconstruidos. Ejemplos de estas operaciones de filtrado incluyen la filtración de desbloqueo, la filtración de

desplazamiento adaptativo de muestras (SAO, por sus siglas en inglés) y la filtración de bucle adaptativa (ALF, por sus siglas en inglés). Los parámetros para estas operaciones de filtrado pueden determinarse mediante un codificador de video y señalizarse explícitamente en el flujo de bits de video codificado o pueden determinarse implícitamente mediante un decodificador de video sin necesidad de que los parámetros se señalicen explícitamente en el flujo de bits de video codificado.

Las técnicas de esta divulgación están dirigidas a la interpredicción en códecs de video, y, más específicamente, a la señalización de imágenes de referencia y la gestión de listas de imágenes de referencia para bloques intercodificados. Con las técnicas existentes, un decodificador de video usa señalización de alto nivel, como a nivel de imagen o de corte, para mantener listas de imágenes de referencia para la interpredicción. Sin embargo, estas técnicas no incluyen adaptaciones a nivel de bloque a las listas de imágenes de referencia, como el reordenamiento. Incluir señalización adicional para permitir dicha adaptación a nivel de bloque puede ser indeseable debido a que el aumento de la sobrecarga de señalización supera las ganancias de codificación logradas por la adaptación de la lista de imágenes de referencia. Esta divulgación describe técnicas para realizar la adaptación de listas de imágenes de referencia a nivel de bloque sin aumentar la sobrecarga de

señalización. Específicamente, un decodificador de video puede configurarse para realizar un proceso de reordenamiento de imágenes de referencia adaptable para generar un nuevo ordenamiento de las imágenes de referencia en una lista de
5 imágenes de referencia sin sobrecarga de señalización. Al reordenar las imágenes de referencia en la lista de imágenes de referencia usando un proceso de reordenamiento de lista de imágenes de referencia adaptativo, los bits necesarios para señalar qué imagen de referencia en la lista de
10 imágenes de referencia se usará para la interpredicción pueden reducirse sin aumentar la distorsión.

Como se usa en esta divulgación, el término codificación de video se refiere genéricamente a codificación de video o decodificación de video. De manera
15 similar, el término codificador de video puede referirse genéricamente a un codificador de video o a un decodificador de video. Además, ciertas técnicas descritas en esta divulgación con respecto a la decodificación de video también pueden aplicarse a la codificación de video, y
20 viceversa. Por ejemplo, los codificadores de video y decodificadores de video están configurados a menudo para realizar el mismo proceso, o procesos recíprocos. Además, un codificador de video generalmente realiza decodificación de video (también llamada reconstrucción) como parte de los
25 procesos para determinar cómo codificar datos de video. Por ejemplo, las técnicas de gestión de lista de imágenes de

referencia a nivel de bloque descritas en esta divulgación pueden ser realizadas tanto por un codificador de video como por un decodificador de video, de manera que tanto el codificador de video como el decodificador de video
5 mantengan las mismas listas de imágenes de referencia con poca o ninguna sobrecarga de señalización.

La Figura 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de video de ejemplo 100 que puede realizar las técnicas de esta
10 divulgación. Las técnicas de esta divulgación se refieren generalmente a codificar (codificar y/o decodificar) datos de video. En general, los datos de video incluyen cualquier dato para procesar un video. Por lo tanto, los datos de video pueden incluir video sin procesar, video codificado,
15 video decodificado (por ejemplo, reconstruido) y metadatos de video, tales como datos de señalización.

Como se muestra en la Figura 1, el sistema 100 incluye un dispositivo fuente 102 que proporciona datos de video codificados que han de ser decodificados y
20 presentados por un dispositivo de destino 116, en este ejemplo. En particular, el dispositivo fuente 102 proporciona los datos de video al dispositivo de destino 116 a través de un medio legible por computador 110. El dispositivo fuente 102 y el dispositivo de destino 116
25 pueden comprender cualquiera de una amplia gama de dispositivos, incluidos computadores de escritorio,

computadores portátiles, dispositivos móviles, tabletas, decodificadores, terminales telefónicos tales como teléfonos inteligentes, televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión de video, dispositivos receptores de difusión, o similares. En algunos casos, el dispositivo fuente 102 y el dispositivo de destino 116 pueden estar equipados para la comunicación inalámbrica y, por lo tanto, pueden denominarse dispositivos de comunicación inalámbrica.

En el ejemplo de la Figura 1, el dispositivo fuente 102 incluye la fuente de video 104, la memoria 106, el codificador de video 200 y la interfaz de salida 108. El dispositivo de destino 116 incluye la interfaz de entrada 122, el decodificador de video 300, la memoria 120 y el dispositivo de visualización 118. De acuerdo con esta divulgación, el codificador de video 200 del dispositivo fuente 102 y el decodificador de video 300 del dispositivo de destino 116 pueden configurarse para aplicar las técnicas para la adaptación de lista de imágenes de referencia de nivel de bloque como se describe en la presente. Por lo tanto, el dispositivo fuente 102 representa un ejemplo de un dispositivo de codificación de video, mientras que el dispositivo de destino 116 representa un ejemplo de un dispositivo de decodificación de video. En otros ejemplos, un dispositivo fuente y un dispositivo de destino pueden

incluir otros componentes o disposiciones. Por ejemplo, el dispositivo fuente 102 puede recibir datos de video desde una fuente de video externa, tal como una cámara externa. De la misma manera, el dispositivo de destino 116 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

El sistema 100 como se muestra en la Figura 1 es simplemente un ejemplo. En general, cualquier dispositivo de codificación y/o decodificación de video digital puede realizar técnicas para la adaptación de listas de imágenes de referencia a nivel de bloque como se describe en la presente. El dispositivo fuente 102 y el dispositivo de destino 116 son simplemente ejemplos de tales dispositivos de codificación en los que el dispositivo fuente 102 genera datos de video codificados para su transmisión al dispositivo de destino 116. Esta divulgación se refiere a un dispositivo de “codificación” como un dispositivo que realiza la codificación (codificación y/o decodificación) de datos. Por lo tanto, el codificador de video 200 y el decodificador de video 300 representan ejemplos de dispositivos de codificación, en particular, un codificador de video y un decodificador de video, respectivamente. En algunos ejemplos, el dispositivo fuente 102 y el dispositivo de destino 116 pueden operar de una manera sustancialmente simétrica de tal manera que cada uno del dispositivo fuente 102 y el dispositivo de destino 116 incluye componentes de

codificación y decodificación de video. Por lo tanto, el sistema 100 puede soportar transmisión de video unidireccional o bidireccional entre el dispositivo fuente 102 y el dispositivo de destino 116, por ejemplo, para 5 transmisión de video, reproducción de video, difusión de video o videotelefonía.

En general, la fuente de video 104 representa una fuente de datos de video (por ejemplo, datos de video sin procesar) y proporciona una serie secuencial de imágenes 10 (también denominadas “cuadros”) de los datos de video al codificador de video 200, que codifica datos para las imágenes. La fuente de video 104 del dispositivo fuente 102 puede incluir un dispositivo de captura de video, tal como una cámara de video, un archivo de video que contiene video 15 sin procesar previamente capturado, y/o una interfaz de alimentación de video para recibir video de un proveedor de contenido de video. Como alternativa adicional, la fuente de video 104 puede generar datos basados en gráficos informáticos como el video fuente, o una combinación de 20 video en directo, video archivado y video generado por computador. En cada caso, el codificador de video 200 codifica los datos de video capturados, precapturados o generados por computador. El codificador de video 200 puede reorganizar las imágenes del orden recibido (a veces denominado “orden de visualización”) en un orden de 25 codificación para la codificación. El codificador de video

200 puede generar un flujo de bits que incluye datos de video codificados. El dispositivo fuente 102 puede entonces emitir los datos de video codificados a través de la interfaz de salida 108 en un medio legible por computador
5 110 para recepción y/o recuperación por, por ejemplo, la interfaz de entrada 122 del dispositivo de destino 116.

La memoria 106 del dispositivo fuente 102 y la memoria 120 del dispositivo de destino 116 representan memorias de propósito general. En algunos ejemplos, las
10 memorias 106, 120 pueden almacenar datos de video sin procesar, por ejemplo, video sin procesar procedente de la fuente de video 104 y datos de video decodificados sin procesar del decodificador de video 300. Adicional o
alternativamente, las memorias 106, 120 pueden almacenar
15 instrucciones de *software* ejecutables por, por ejemplo, el codificador de video 200 y el decodificador de video 300, respectivamente. Aunque la memoria 106 y la memoria 120 se muestran por separado del codificador de video 200 y del decodificador de video 300 en este ejemplo, debe entenderse
20 que el codificador de video 200 y el decodificador de video 300 también pueden incluir memorias internas para fines funcionalmente similares o equivalentes. Además, las memorias 106, 120 pueden almacenar datos de video codificados, por ejemplo, salida desde el codificador de
25 video 200 y entrada al decodificador de video 300. En algunos ejemplos, partes de las memorias 106, 120 pueden

asignarse como uno o más búferes de video, por ejemplo, para almacenar datos de video sin procesar, decodificados y/o codificados.

El medio legible por computador 110 puede
5 representar cualquier tipo de medio o dispositivo capaz de transportar los datos de video codificados desde el dispositivo fuente 102 al dispositivo de destino 116. En un ejemplo, el medio legible por computador 110 representa un medio de comunicación para permitir que el dispositivo fuente
10 102 transmita datos de video codificados directamente al dispositivo de destino 116 en tiempo real, por ejemplo, a través de una red de radiofrecuencia o una red basada en computador. La interfaz de salida 108 puede modular una señal de transmisión que incluye los datos de video codificados, y
15 la interfaz de entrada 122 puede demodular la señal de transmisión recibida, de acuerdo con un estándar de comunicación, tal como un protocolo de comunicación inalámbrica. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o por cable, tal
20 como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global como Internet. El medio de comunicación puede incluir enrutadores,
25 conmutadores, estaciones base o cualquier otro equipo que

pueda ser útil para facilitar la comunicación desde el dispositivo fuente 102 al dispositivo de destino 116.

En algunos ejemplos, el dispositivo fuente 102 puede emitir datos codificados desde la interfaz de salida 108 al dispositivo de almacenamiento 112. De manera similar, el dispositivo de destino 116 puede acceder a datos codificados desde el dispositivo de almacenamiento 112 a través de la interfaz de entrada 122. El dispositivo de almacenamiento 112 puede incluir cualquiera de una variedad de medios de almacenamiento de datos distribuidos o a los que se accede localmente, tales como un disco duro, discos Blu-ray, DVD, CD-ROM, memoria flash, memoria volátil o no volátil, o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de video codificados.

En algunos ejemplos, el dispositivo fuente 102 puede emitir datos de video codificados al servidor de archivos 114 u otro dispositivo de almacenamiento intermedio que puede almacenar los datos de video codificados generados por el dispositivo fuente 102. El dispositivo de destino 116 puede acceder a datos de video almacenados desde el servidor de archivos 114 a través de transmisión o descarga.

El servidor de archivos 114 puede ser cualquier tipo de dispositivo servidor capaz de almacenar datos de video codificados y transmitir esos datos de video codificados al dispositivo de destino 116. El servidor de archivos 114 puede representar un servidor web (por

ejemplo, para un sitio web), un servidor configurado para proporcionar un servicio de protocolo de transferencia de archivos (tal como protocolo de transferencia de archivos (FTP, por sus siglas en inglés) o protocolo de entrega de archivos a través de transporte unidireccional (FLUTE, por sus siglas en inglés)), un dispositivo de red de entrega de contenido (CDN, por sus siglas en inglés), un servidor de protocolo de transferencia de hipertexto (HTTP, por sus siglas en inglés), un servicio de difusión multidifusión multimedia (MBMS, por sus siglas en inglés), y/o un dispositivo de almacenamiento conectado a la red (NAS, por sus siglas en inglés). Adicional o alternativamente, el servidor de archivos 114 puede implementar uno o más protocolos de transmisión HTTP, tales como transmisión dinámica adaptativa sobre HTTP (DASH, por sus siglas en inglés), transmisión en vivo HTTP (HLS, por sus siglas en inglés), protocolo de transmisión en tiempo real (RTSP, por sus siglas en inglés), transmisión dinámica HTTP, o similares.

El dispositivo de destino 116 puede acceder a datos de video codificados desde el servidor de archivos 114 a través de cualquier conexión de datos estándar, incluida una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión por cable (por ejemplo, línea de abonado digital (DSL, por sus siglas en inglés), módem por cable, etc.),

o una combinación de ambos que sea adecuada para acceder a datos de video codificados almacenados en el servidor de archivos 114. La interfaz de entrada 122 puede configurarse para funcionar de acuerdo con uno o más de los diversos
5 protocolos analizados anteriormente para recuperar o recibir datos de medios desde el servidor de archivos 114, u otros protocolos de este tipo para recuperar datos de medios.

La interfaz de salida 108 y la interfaz de entrada
10 122 pueden representar transmisores/receptores inalámbricos, módems, componentes de red cableados (por ejemplo, tarjetas Ethernet), componentes de comunicación inalámbrica que operan de acuerdo con cualquiera de una variedad de estándares IEEE 802.11 u otros componentes físicos. En ejemplos donde la
15 interfaz de salida 108 y la interfaz de entrada 122 comprenden componentes inalámbricos, la interfaz de salida 108 y la interfaz de entrada 122 pueden configurarse para transferir datos, tales como datos de video codificados, de acuerdo con un estándar de comunicación celular, tal como 4G, 4G-LTE
20 (Evolución a largo plazo), LTE Avanzada, 5G, o similares. En algunos ejemplos donde la interfaz de salida 108 comprende un transmisor inalámbrico, la interfaz de salida 108 y la interfaz de entrada 122 pueden configurarse para transferir datos, tales como datos de video codificados, de acuerdo con
25 otros estándares inalámbricos, tales como una especificación IEEE 802.11, una especificación IEEE 802.15 (por ejemplo,

ZigBee™), un estándar Bluetooth™, o similares. En algunos ejemplos, el dispositivo fuente 102 y/o el dispositivo de destino 116 pueden incluir dispositivos respectivos de sistema en un chip (SoC, por sus siglas en inglés). Por ejemplo, el dispositivo fuente 102 puede incluir un dispositivo SoC para realizar la funcionalidad atribuida al codificador de video 200 y/o a la interfaz de salida 108, y el dispositivo de destino 116 puede incluir un dispositivo SoC para realizar la funcionalidad atribuida al decodificador de video 300 y/o a la interfaz de entrada 122.

Las técnicas de esta divulgación pueden aplicarse a la codificación de video en soporte de cualquiera de una variedad de aplicaciones multimedia, tales como transmisiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de video en flujo continuo por Internet, tales como transmisión dinámica adaptativa sobre HTTP (DASH), video digital que se codifica en un medio de almacenamiento de datos, decodificación de video digital almacenado en un medio de almacenamiento de datos u otras aplicaciones.

La interfaz de entrada 122 del dispositivo de destino 116 recibe un flujo de bits de video codificado desde un medio legible por computador 110 (por ejemplo, un medio de comunicación, un dispositivo de almacenamiento 112, un servidor de archivos 114 o similares). El flujo de bits de video codificado puede incluir información de

señalización definida por el codificador de video 200, que también es usado por el decodificador de video 300, tal como elementos de sintaxis que tienen valores que describen características y/o procesamiento de bloques de video u
5 otras unidades codificadas (por ejemplo, cortes, imágenes, grupos de imágenes, secuencias, o similares). El dispositivo de visualización 118 muestra imágenes decodificadas de los datos de video decodificados a un usuario. El dispositivo de visualización 118 puede representar cualquiera de una
10 variedad de dispositivos de visualización tales como una pantalla de cristal líquido (LCD, por sus siglas en inglés), una pantalla de plasma, una pantalla de diodo emisor de luz orgánico (OLED, por sus siglas en inglés) u otro tipo de dispositivo de visualización.

15 Aunque no se muestra en la Figura 1, en algunos ejemplos, cada uno del codificador de video 200 y el decodificador de video 300 puede estar integrado con un codificador de audio y/o decodificador de audio, y puede incluir unidades MUX-DEMUX apropiadas, u otro *hardware* y/o
20 *software*, para manejar flujos multiplexados que incluyen tanto audio como video en un flujo de datos común.

Cada uno del codificador de video 200 y el decodificador de video 300 puede implementarse como cualquiera de una variedad de circuitos de codificador y/o
25 decodificador adecuados, tales como uno o más microprocesadores, procesadores de señal digital (DSP, por

sus siglas en inglés), circuitos integrados de aplicación específica (ASIC, por sus siglas en inglés), matrices de puertas programables en campo (FPGA, por sus siglas en inglés), 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 legible por computador adecuado, no transitorio y ejecutar las instrucciones en *hardware* usando uno o más procesadores para realizar las técnicas de esta descripción. Cada uno del codificador de video 200 y del decodificador de video 300 puede incluirse en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador combinado (CODEC, por sus siglas en inglés) en un dispositivo respectivo. Un dispositivo que incluye el codificador de video 200 y/o el decodificador de video 300 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

El codificador de video 200 y el decodificador de video 300 pueden funcionar de acuerdo con un estándar de codificación de video, tal como ITU-T H.265, también denominado Codificación de video de alta eficiencia (HEVC) o extensiones de este, tales como las extensiones de codificación de video multivista y/o escalable.

Alternativamente, el codificador de video 200 y el decodificador de video 300 pueden funcionar de acuerdo con otros estándares patentados o de la industria, tales como ITU-T H.266, también denominado codificación versátil de video (VVC). En otros ejemplos, el codificador de video 200 y el decodificador de video 300 pueden funcionar de acuerdo con un códec/formato de video patentado, tal como AOMedia Video 1 (AV1, por sus siglas en inglés), extensiones de AV1 y/o versiones sucesivas de AV1 (por ejemplo, AV2). En otros ejemplos, el codificador de video 200 y el decodificador de video 300 pueden funcionar de acuerdo con otros formatos patentados o estándares de la industria. Sin embargo, las técnicas de esta divulgación no se limitan a ningún estándar o formato de codificación particular. En general, el codificador de video 200 y el decodificador de video 300 pueden configurarse para realizar las técnicas de esta descripción junto con cualquier técnica de codificación de video que use listas de imágenes de referencia para la interpredicción.

En general, el codificador de video 200 y el decodificador de video 300 pueden realizar la codificación basada en bloques de imágenes. El término “bloque” generalmente se refiere a una estructura que incluye datos a ser procesados (por ejemplo, codificados, decodificados o usados de cualquier otra manera en el proceso de codificación y/o decodificación). Por ejemplo, un bloque

puede incluir una matriz bidimensional de muestras de luminancia y/o datos de crominancia. En general, el codificador de video 200 y el decodificador de video 300 pueden codificar datos de video representados en un formato YUV (por ejemplo, Y, Cb, Cr). Es decir, en lugar de codificar datos rojos, verdes y azules (RGB, por sus siglas en inglés) para muestras de una imagen, el codificador de video 200 y el decodificador de video 300 pueden codificar componentes de luminancia y crominancia, donde los componentes de crominancia pueden incluir componentes de crominancia tanto de tono rojo como de tono azul. En algunos ejemplos, el codificador de video 200 convierte los datos con formato RGB recibidos en una representación YUV antes de la codificación, y el decodificador de video 300 convierte la representación YUV al formato RGB. Alternativamente, las unidades de procesamiento previo y posterior (no mostradas) pueden realizar estas conversiones.

Esta divulgación puede referirse generalmente a la codificación (por ejemplo, codificación y decodificación) de imágenes para incluir el proceso de codificación o decodificación de datos de la imagen. De manera similar, esta divulgación puede referirse a la codificación de bloques de una imagen para incluir el proceso de codificación o decodificación de datos para los bloques, por ejemplo, predicción y/o codificación

residual. Un flujo de bits de video codificado generalmente incluye una serie de valores para elementos de sintaxis representativos de decisiones de codificación (por ejemplo, modos de codificación) y partición de imágenes en bloques. Por lo tanto, las referencias a la codificación de una imagen o un bloque deben entenderse generalmente como valores de codificación para elementos de sintaxis que forman la imagen o bloque.

HEVC define varios bloques, incluidas unidades de codificación (CU), unidades de predicción (PU, por sus siglas en inglés) y unidades de transformada (TU, por sus siglas en inglés). De acuerdo con HEVC, un codificador de video (como el codificador de video 200) divide una unidad de árbol de codificación (CTU) en CU de acuerdo con una estructura de árbol cuádruple. Es decir, el codificador de video divide las CTU y las CU en cuatro cuadrados iguales, no superpuestos, y cada nodo del árbol cuádruple tiene cero o cuatro nodos secundarios. Los nodos sin nodos secundarios pueden denominarse “nodos hoja”, y las CU de dichos nodos hoja pueden incluir una o más PU y/o una o más TU. El codificador de video puede además particionar las PU y las TU. Por ejemplo, en HEVC, un cuádruple residual (RQT, por sus siglas en inglés) representa la partición de las TU. En HEVC, las PU representan datos de interpredicción, mientras que las TU representan datos residuales. Las CU que se

predijeron dentro de las predicciones incluyen información de intrapredicción, como una indicación intramodo.

Como otro ejemplo, el codificador de video 200 y el decodificador de video 300 pueden configurarse para funcionar de acuerdo con VVC. De acuerdo con VVC, un codificador de video (tal como el codificador de video 200) divide una imagen en una pluralidad de unidades de árbol de codificación (CTU). El codificador de video 200 puede particionar una CTU de acuerdo con una estructura de árbol, tal como una estructura de árbol cuaternario-binario (QTBT, por sus siglas en inglés) o una estructura de árbol multitipo (MTT, por sus siglas en inglés). La estructura QTBT elimina los conceptos de múltiples tipos de partición, como la separación entre CU, PU y TU de HEVC. Una estructura QTBT incluye dos niveles: un primer nivel particionado de acuerdo con la partición de árbol cuaternario, y un segundo nivel particionado de acuerdo con la partición de árbol binario. Un nodo raíz de la estructura QTBT corresponde a una CTU. Los nodos hoja de los árboles binarios corresponden a unidades de codificación (CU).

En una estructura de partición MTT, los bloques pueden particionarse usando una partición de árbol cuaternario (QT, por sus siglas en inglés), una partición de árbol binario (BT, por sus siglas en inglés) y uno o más tipos de particiones de árbol triple (TT, por sus siglas

en inglés) (también llamado árbol ternario (TT)). Una partición de árbol triple o ternario es una partición donde un bloque se divide en tres subbloques. En algunos ejemplos, una partición de árbol triple o ternario divide un bloque en tres subbloques sin dividir el bloque original a través del centro. Los tipos de partición en MTT (por ejemplo, QT, BT y TT) pueden ser simétricos o asimétricos.

Cuando se opera de acuerdo con el códec AV1, el codificador de video 200 y el decodificador de video 300 pueden configurarse para codificar datos de video en bloques. En AV1, el bloque de codificación más grande que se puede procesar se llama superbloque. En AV1, un superbloque puede ser muestras de 128x128 luma o muestras de 64x64 luma. Sin embargo, en los formatos de codificación de video sucesores (por ejemplo, AV2), un superbloque puede definirse por diferentes tamaños de muestra de luma (por ejemplo, más grandes). En algunos ejemplos, un superbloque es el nivel superior de un árbol cuádruple de bloques. El codificador de video 200 puede dividir además un superbloque en bloques de codificación más pequeños. El codificador de video 200 puede dividir un superbloque y otros bloques de codificación en bloques más pequeños usando particiones cuadradas o no cuadradas. Los bloques no cuadrados pueden incluir bloques $N/2 \times N$, $N \times N/2$, $N/4 \times N$ y $N \times N/4$. El codificador de video 200 y el decodificador de video 300 pueden realizar procesos

separados de predicción y transformada en cada uno de los bloques de codificación.

AV1 también define un mosaico de datos de video. Un mosaico es una matriz rectangular de superbloques que pueden codificarse independientemente de otros mosaicos. Es decir, el codificador de video 200 y el decodificador de video 300 pueden codificar y decodificar, respectivamente, bloques de codificación dentro de un mosaico sin usar datos de video de otros mosaicos. Sin embargo, el codificador de video 200 y el decodificador de video 300 pueden realizar el filtrado a través de los límites del mosaico. Los mosaicos pueden ser uniformes o no uniformes en tamaño. La codificación basada en mosaico puede permitir el procesamiento paralelo y/o subprocesos múltiples para implementaciones de codificador y decodificador.

En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden usar una única estructura QTBT o MTT para representar cada uno de los componentes de luminancia y crominancia, mientras que, en otros ejemplos, el codificador de video 200 y el decodificador de video 300 pueden usar dos o más estructuras QTBT o MTT, tales como una estructura QTBT/MTT para el componente de luminancia y otra estructura QTBT/MTT para ambos componentes de crominancia (o dos estructuras QTBT/MTT para los respectivos componentes de crominancia).

El codificador de video 200 y el decodificador de video 300 pueden configurarse para usar particiones de árbol cuádruple, partición QTBT, partición MTT, partición de superbloque u otras estructuras de partición.

5 En algunos ejemplos, una CTU incluye un bloque de árbol de codificación (CTB) de muestras de luma, dos CTB correspondientes de muestras de croma de una imagen que tiene tres matrices de muestras, o un CTB de muestras de una imagen monocromática o una imagen que se codifica usando tres planos
10 de color separados y estructuras de sintaxis usadas para codificar las muestras. Un CTB puede ser un bloque $N \times N$ de muestras para algún valor de N tal que la división de un componente en CTB es una partición. Un componente es una matriz o muestra única de una de las tres matrices (luma y
15 dos croma) que componen una imagen en formato de color 4:2:0, 4:2:2, o 4:4:4 o la matriz o una sola muestra de la matriz que compone una imagen en formato monocromático. En algunos ejemplos, un bloque de codificación es un bloque $M \times N$ de muestras para algunos valores de M y N tal que una división
20 de un CTB en bloques de codificación es una partición.

Los bloques (por ejemplo, CTU o CU) pueden agruparse de diversas maneras en una imagen. Como ejemplo, un bloque puede referirse a una región rectangular de filas CTU dentro de un mosaico particular en una imagen. Un
25 mosaico puede ser una región rectangular de CTU dentro de una columna de mosaico particular y una fila de mosaico

particular en una imagen. Una columna de mosaico se refiere a una región rectangular de CTU que tiene una altura igual a la altura de la imagen y un ancho especificado por elementos de sintaxis (por ejemplo, en un conjunto de parámetros de imagen). Una fila de mosaico se refiere a una región rectangular de CTU que tiene una altura especificada por elementos de sintaxis (por ejemplo, en un conjunto de parámetros de imagen) y un ancho igual al ancho de la imagen.

En algunos ejemplos, un mosaico puede dividirse en múltiples bloques, cada uno de los cuales puede incluir una o más filas de CTU dentro del mosaico. Un mosaico que no está particionado en múltiples bloques también puede denominarse bloque. Sin embargo, un bloque que es un verdadero subconjunto de un mosaico no puede denominarse mosaico. Los bloques en una imagen también pueden estar dispuestos en un corte. Un corte puede ser un número entero de bloques de una imagen que puede estar contenido exclusivamente en una única unidad de capa de abstracción de red (NAL, por sus siglas en inglés). En algunos ejemplos, un corte incluye un número de mosaicos completos o solo una secuencia consecutiva de bloques completos de un mosaico.

Esta divulgación puede usar “NxN” y “N por N” indistintamente para referirse a las dimensiones de muestra de un bloque (tal como una CU u otro bloque de video) en

términos de dimensiones verticales y horizontales, por ejemplo, muestras 16x16 o muestras 16 por 16. En general, una CU 16x16 tendrá 16 muestras en una dirección vertical ($y = 16$) y 16 muestras en una dirección horizontal ($x = 16$). De la misma manera, una CU NxN generalmente tiene N muestras en una dirección vertical y N muestras en una dirección horizontal, donde N representa un valor entero no negativo. Las muestras en una CU pueden estar dispuestas en filas y columnas. Además, las CU no necesitan tener necesariamente el mismo número de muestras en la dirección horizontal que en la dirección vertical. Por ejemplo, las CU pueden comprender muestras de NxM, donde M no es necesariamente igual a N.

El codificador de video 200 codifica datos de video para las CU que representan predicción y/o información residual, y otra información. La información de predicción indica cómo se predecirá la CU para formar un bloque de predicción para la CU. La información residual generalmente representa las diferencias muestra por muestra entre las muestras de la CU antes de la codificación y el bloque de predicción.

Para predecir una CU, el codificador de video 200 generalmente puede formar un bloque de predicción para la CU a través de la interpredicción o intrapredicción. La interpredicción generalmente se refiere a predecir la CU a partir de datos de una imagen previamente codificada,

mientras que la intrapredicción generalmente se refiere a predecir la CU a partir de datos previamente codificados de la misma imagen. Para realizar la interpredicción, el codificador de video 200 puede generar el bloque de predicción usando uno o más vectores de movimiento. El codificador de video 200 puede realizar generalmente una búsqueda de movimiento para identificar un bloque de referencia que coincide estrechamente con la CU, por ejemplo, en términos de diferencias entre la CU y el bloque de referencia. El codificador de video 200 puede calcular una métrica de diferencia usando una suma de diferencia absoluta (SAD, por sus siglas en inglés), suma de diferencias cuadradas (SSD, por sus siglas en inglés), diferencia absoluta media (MAD, por sus siglas en inglés), diferencias cuadradas medias (MSD, por sus siglas en inglés) u otros cálculos de diferencia similares para determinar si un bloque de referencia coincide estrechamente con la CU actual. En algunos ejemplos, el codificador de video 200 puede predecir la CU actual usando predicción unidireccional o predicción bidireccional.

Algunos ejemplos de VVC también proporcionan un modo de compensación de movimiento afín, que puede considerarse un modo de interpredicción. En el modo de compensación de movimiento afín, el codificador de video 200 puede determinar dos o más vectores de movimiento que representan movimiento no traslacional, como acercamiento o

alejamiento, rotación, movimiento de perspectiva u otros tipos de movimiento irregular.

Para realizar la intrapredicción, el codificador de video 200 puede seleccionar un modo de intrapredicción para generar el bloque de predicción. Algunos ejemplos de VVC proporcionan sesenta y siete modos de intrapredicción, incluidos varios modos direccionales, así como el modo plano y el modo DC. En general, el codificador de video 200 selecciona un modo de intrapredicción que describe muestras vecinas a un bloque actual (por ejemplo, un bloque de una CU) a partir del cual predecir muestras del bloque actual. Dichas muestras generalmente pueden estar arriba, arriba y a la izquierda, o a la izquierda del bloque actual en la misma imagen que el bloque actual, suponiendo que el codificador de video 200 codifica CTU y CU en orden de escaneo ráster (de izquierda a derecha, de arriba a abajo).

El codificador de video 200 codifica datos que representan el modo de predicción para un bloque actual. Por ejemplo, para los modos de interpredicción, el codificador de video 200 puede codificar datos que representan cuál de los diversos modos de interpredicción disponibles se usa, así como información de movimiento para el modo correspondiente. Para la interpredicción unidireccional o bidireccional, por ejemplo, el codificador de video 200 puede codificar vectores de movimiento usando predicción de vectores de movimiento avanzada (AMVP, por

sus siglas en inglés) o modo de fusión. El codificador de video 200 puede usar modos similares para codificar vectores de movimiento para el modo de compensación de movimiento afín.

5 Para bloques codificados en modo de fusión, el codificador de video 200 y el decodificador de video 300 pueden configurarse para mantener una lista de candidatos, con cada candidato en la lista que incluye un conjunto completo de información de movimiento (por ejemplo, un
10 vector de movimiento, una dirección de movimiento y un índice de imagen de referencia). Para bloques codificados en modo AMVP, el codificador de video 200 y el decodificador de video 300 pueden configurarse para mantener de manera similar una lista de candidatos, pero
15 en lugar de información de movimiento completa, cada candidato en la lista solo incluye un predictor de vector de movimiento (MVP, por sus siglas en inglés). El decodificador de video 300 puede recibir por separado una diferencia de vector de movimiento (MVD, por sus siglas en
20 inglés) y determinar el vector de movimiento para predecir un bloque como el MVP más la MVD.

AV1 incluye dos técnicas generales para codificar y decodificar un bloque de codificación de datos de video. Las dos técnicas generales son la intrapredicción (por
25 ejemplo, predicción intracuadro o predicción espacial) y la interpredicción (por ejemplo, predicción intracuadro o

predicción temporal). En el contexto de AV1, cuando se predican bloques de un cuadro actual de datos de video usando un modo de intrapredicción, el codificador de video 200 y el decodificador de video 300 no usan datos de video de otros cuadros de datos de video. Para la mayoría de los modos de intrapredicción, el codificador de video 200 codifica bloques de un cuadro actual basado en la diferencia entre los valores de muestra en el bloque actual y los valores predichos generados a partir de muestras de referencia en el mismo cuadro. El codificador de video 200 determina los valores predichos generados a partir de las muestras de referencia basado en el modo de intrapredicción.

Después de la predicción, tal como la intrapredicción o la interpredicción de un bloque, el codificador de video 200 puede calcular datos residuales para el bloque. El dato residual, como un bloque residual, representa las diferencias muestra por muestra entre el bloque y un bloque de predicción para el bloque, formado usando el modo de predicción correspondiente. El codificador de video 200 puede aplicar una o más transformadas al bloque residual, para producir datos transformados en un dominio de transformada en lugar del dominio de muestra. Por ejemplo, el codificador de video 200 puede aplicar una transformada de coseno discreta (DCT, por sus siglas en inglés), una transformada entera, una transformada de ondícula o una transformada conceptualmente similar a datos de video

residuales. Adicionalmente, el codificador de video 200 puede aplicar una transformada secundaria después de la primera transformada, tal como una transformada secundaria no separable dependiente del modo (MDNST, por sus siglas en inglés), una transformada dependiente de señal, una transformada de Karhunen-Loeve (KLT, por sus siglas en inglés), o similares. El codificador de video 200 produce coeficientes de transformada después de la aplicación de una o más transformadas.

10 Como se señaló anteriormente, después de cualquier transformada para producir coeficientes de transformada, el codificador de video 200 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación generalmente se refiere a un proceso en el que los
15 coeficientes de transformada se cuantifican para posiblemente reducir la cantidad de datos usados para representar los coeficientes de transformada, proporcionando una compresión adicional. Al realizar el proceso de cuantificación, el codificador de video 200 puede reducir la profundidad de bits
20 asociada con algunos o todos los coeficientes de transformada. Por ejemplo, el codificador de video 200 puede redondear un valor de n bits a un valor de m bits durante la cuantificación, donde n es mayor que m . En algunos ejemplos, para realizar la cuantificación, el codificador de video 200
25 puede realizar un desplazamiento a la derecha bit a la derecha del valor a cuantificar.

Después de la cuantificación, el codificador de video 200 puede escanear los coeficientes de transformada, al producir un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformada cuantificados. El escaneo puede estar diseñado para colocar coeficientes de transformada de mayor energía (y por lo tanto de menor frecuencia) en la parte frontal del vector y para colocar coeficientes de transformada de energía más baja (y por lo tanto de mayor frecuencia) en la parte posterior del vector. En algunos ejemplos, el codificador de video 200 puede usar un orden de exploración predefinido para escanear los coeficientes de transformada cuantificados para producir un vector serializado, y después codificar por entropía los coeficientes de transformada cuantificados del vector. En otros ejemplos, el codificador de video 200 puede realizar un escaneo adaptativo. Después de escanear los coeficientes de transformada cuantificados para formar el vector unidimensional, el codificador de video 200 puede codificar por entropía el vector unidimensional, por ejemplo, de acuerdo con la codificación aritmética binaria adaptativa al contexto (CABAC). El codificador de video 200 también puede codificar por entropía valores para elementos de sintaxis que describen metadatos asociados con los datos de video codificados para el uso mediante el decodificador de video 300 en la decodificación de los datos de video.

Para realizar CABAC, el codificador de video 200 puede asignar un contexto dentro de un modelo de contexto a un símbolo a transmitir. El contexto puede relacionarse, por ejemplo, con si los valores vecinos del símbolo son de valor cero o no. La determinación de probabilidad puede basarse en un contexto asignado al símbolo.

El codificador de video 200 puede generar además datos de sintaxis, tales como datos de sintaxis basados en bloques, datos de sintaxis basados en imágenes y datos de sintaxis basados en secuencia, para el decodificador de video 300, por ejemplo, en un encabezado de imagen, un encabezado de bloque, un encabezado de corte u otros datos de sintaxis, tales como un conjunto de parámetros de secuencia (SPS, por sus siglas en inglés), un conjunto de parámetros de imagen (PPS, por sus siglas en inglés) o un conjunto de parámetros de video (VPS, por sus siglas en inglés). El decodificador de video 300 también puede decodificar tales datos de sintaxis para determinar cómo decodificar los datos de video correspondientes.

De esta manera, el codificador de video 200 puede generar un flujo de bits que incluye datos de video codificados, por ejemplo, elementos de sintaxis que describen la partición de una imagen en bloques (por ejemplo, CU) y predicción y/o información residual para los bloques. En última instancia, el decodificador de video 300 puede recibir el flujo de bits y decodificar los datos de video codificados.

En general, el decodificador de video 300 realiza un proceso recíproco al realizado por el codificador de video 200 para decodificar los datos de video codificados del flujo de bits. Por ejemplo, el decodificador de video 300 puede decodificar valores para elementos de sintaxis del flujo de bits usando CABAC de una manera sustancialmente similar, aunque recíproca a, el proceso de codificación CABAC del codificador de video 200. Los elementos de sintaxis pueden definir información de partición para la partición de una imagen en CTU, y la partición de cada CTU de acuerdo con una estructura de partición correspondiente, tal como una estructura QTBT, para definir las CU de la CTU. Los elementos de sintaxis pueden definir adicionalmente predicción e información residual para bloques (por ejemplo, CU) de datos de video.

La información residual puede estar representada por, por ejemplo, coeficientes de transformada cuantificados. El decodificador de video 300 puede cuantizar de manera inversa y transformar de manera inversa los coeficientes de transformada cuantificados de un bloque para reproducir un bloque residual para el bloque. El decodificador de video 300 usa un modo de predicción señalizado (intra o interpredicción) e información de predicción relacionada (por ejemplo, información de movimiento para la interpredicción) para formar un bloque de predicción para el bloque. El decodificador de video

300 puede entonces combinar el bloque de predicción y el bloque residual (muestra por muestra) para reproducir el bloque original. El decodificador de video 300 puede realizar un procesamiento adicional, tal como realizar un proceso de desbloqueo para reducir los artefactos visuales a lo largo de los límites del bloque.

Esta divulgación generalmente puede referirse a “señalizar” cierta información, como elementos de sintaxis. El término “señalización” puede referirse generalmente a la comunicación de valores para elementos de sintaxis y/u otros datos usados para decodificar datos de video codificados. Es decir, el codificador de video 200 puede señalar valores para elementos de sintaxis en el flujo de bits. En general, la señalización se refiere a generar un valor en el flujo de bits. Como se ha indicado anteriormente, el dispositivo fuente 102 puede transportar el flujo de bits al dispositivo de destino 116 sustancialmente en tiempo real, o no en tiempo real, tal como podría ocurrir cuando se almacenan elementos de sintaxis en el dispositivo de almacenamiento 112 para su posterior recuperación por el dispositivo de destino 116.

Los códecs de video, como HEVC y VVC, usan la interpredicción para derivar señales de predicción de imágenes de referencia distintas de la imagen codificada actual. Las imágenes de referencia se organizan en una o dos listas de imágenes de referencia. Las dos listas se denotan como Lista 0 y Lista 1. Cada lista de imágenes de referencia

puede incluir una o múltiples imágenes de referencia. El codificador de video 200 codifica un elemento de sintaxis que indica una dirección de interpredicción que identifica si se usa la Lista 0, la Lista 1 o ambas listas para la interpredicción del bloque codificado actual. El codificador de video 200 también codifica otro elemento de sintaxis que indica un índice de referencia (para cada lista de referencia usada) para identificar cuál de las imágenes de referencia en la lista se usa para la interpredicción. En un modo de fusión, el decodificador de video 300 deriva la dirección de interpredicción y el índice de referencia junto con la información del vector de movimiento del candidato de fusión seleccionado. En el modo AMVP, se señalizan elementos de sintaxis adicionales que indican una dirección de interpredicción y un índice de referencia en el flujo de bits para el bloque intercodificado.

Cuando se señala la dirección de interpredicción, el codificador de video 200 señala un primer bin para indicar si el modo de interpredicción (por ejemplo, la dirección de predicción) es un modo de bipredicción o unipredicción. Si el modo es unipredicción, se señala un segundo bin para indicar si se usa la Lista 0 o la Lista 1. Si el modo es b-predicción, entonces no se necesita el bin adicional porque se usan la Lista 0 y la Lista 1. El índice de referencia correspondiente se señala en el flujo de bits después de eso. Cuando se usa la

bipredicción, tanto la Lista 0 como la Lista 1 se usan para la interpredicción, y los índices de referencia para ambas listas deben señalizarse en el flujo de bits. El índice de referencia, que va de 0 a N-1, se binariza usando codificación unaria truncada, siendo N el número de imágenes de referencia en la lista. CABAC puede aplicarse para codificar algunos de los bins.

Tanto HEVC como VVC soportan estructuras flexibles de imagen de referencia, que se pueden señalar como parte de la sintaxis de alto nivel que permite que cada corte/imagen tenga diferentes listas de imágenes de referencia. Es decir, el codificador de video 200 y el decodificador de video 300 se configuran para mantener las mismas listas de imágenes de referencia, con las mismas imágenes y el mismo ordenamiento. Para que el codificador de video 200 y el decodificador de video 300 puedan mantener las mismas listas de imágenes de referencia, el codificador de video 200 incluye alguna señalización en el flujo de bits, el decodificador de video 300 usa la señalización para determinar qué imágenes y en qué orden mantener en las listas de imágenes de referencia. Sin embargo, la flexibilidad no se soporta a nivel de bloque debido a la mayor sobrecarga de señalización que se incurriría para señalar una estructura de imagen de referencia para cada bloque.

En HEVC, la resolución espacial de las imágenes no puede cambiar a menos que una nueva secuencia que utilice

un nuevo SPS comience con una imagen de punto de acceso intraaleatorio (IRAP, por sus siglas en inglés). VVC permite cambios de resolución de imagen dentro de una secuencia en una posición sin codificar una imagen IRAP, que siempre está intracodificada. Esta característica a veces se denomina remuestreo de imágenes de referencia (RPR, por sus siglas en inglés) porque la característica utiliza el remuestreo de una imagen de referencia utilizada para la interpredicción cuando esa imagen de referencia tiene una resolución diferente a la imagen actual que se decodificará. Para evitar pasos de procesamiento adicionales, el proceso de RPR en VVC se incrusta en el proceso de compensación de movimiento y se realiza a nivel de bloque. En la etapa de compensación de movimiento, la relación de escala se usa junto con la información de movimiento para localizar las muestras de referencia en la imagen de referencia que se usará en el proceso de interpolación.

En VVC, la relación de escala se restringe a ser mayor o igual a $1/2$ (2 veces el muestreo descendente de la imagen de referencia a la imagen actual) y menor o igual a 8 (8 veces el muestreo ascendente). Se especifican tres conjuntos de filtros de remuestreo con diferentes cortes de frecuencia para manejar varias relaciones de escala entre una imagen de referencia y la imagen actual. Los tres conjuntos de filtros de remuestreo se aplican

respectivamente para la relación de escala que va de 1/2 a 1/1,75, de 1/1,75 a 1/1,25, y de 1/1,25 a 8. Cada conjunto de filtros de remuestreo tiene 16 fases para luma y 32 fases para croma que es igual al caso de los filtros de interpolación de compensación de movimiento. En un ejemplo, el conjunto de filtros de interpolación MC normal se usa en el caso de una relación de escala que varía de 1/1,25 a 8. El proceso normal de interpolación MC es un caso especial del proceso de remuestreo con una relación de escala que va de 1/1,25 a 8. Además del movimiento de bloque traslacional convencional, el modo afín tiene tres conjuntos de filtros de interpolación de 6 toques que se usan para que el componente luma cubra las diferentes relaciones de escala en RPR. Las relaciones de escala horizontal y vertical se derivan según el ancho y la altura de la imagen, y los desajustes de escala izquierda, derecha, superior e inferior especificados para la imagen de referencia y la imagen actual.

Para soportar esta función, la resolución de imagen y la ventana de conformidad correspondiente se señalizan en el PPS en lugar de en el SPS, mientras que en el SPS se señalizan la resolución de imagen máxima.

Esta divulgación describe técnicas para la adaptación de imágenes de referencia a nivel de bloque. El codificador de video 200 y el decodificador de video 300 pueden configurarse para organizar de forma adaptativa,

por ejemplo, sin sobrecarga de señalización, las imágenes de referencia a nivel de bloque. Es decir, cada bloque puede tener un ordenamiento diferente de imágenes de referencia dentro de una lista de imágenes de referencia.

5 En general, las imágenes de referencia se disponen (reordenan) de manera que se asigna un índice más pequeño (un número menor de bins) a una imagen de referencia (o un conjunto de imágenes de referencia múltiples) que es más probable que se elija para el bloque actual, lo que puede

10 resultar en una reducción de los bits necesarios para señalar un índice de imagen de referencia, y así mejorar el rendimiento general de codificación.

El codificador de video 200 y el decodificador de video 300 pueden configurarse para determinar la probabilidad

15 de un índice de referencia particular en función de un costo de emparejamiento de plantilla. Para una imagen de referencia con mayor probabilidad, se puede asignar un índice más pequeño. Es decir, para una hipótesis de imagen de referencia con un costo de emparejamiento de plantilla más pequeño, el

20 codificador de video 200 y el decodificador de video 300 pueden configurarse para asignar un índice más pequeño a la imagen de referencia asociada que da como resultado el menor costo de emparejamiento de plantilla. El codificador de video 200 y el decodificador de video 300 pueden configurarse para

25 calcular el costo de emparejamiento de plantilla basado en la distorsión entre la señal de reconstrucción de las

muestras vecinas en la plantilla y una señal de predicción generada por la hipótesis de referencia y la información de movimiento correspondiente del bloque actual. Para cada hipótesis de referencia, el codificador de video 200 y el
5 decodificador de video 300 pueden configurarse para:

- 1) Dada la hipótesis de referencia y otros elementos de sintaxis de información de movimiento (por ejemplo, MVP, MVD, resolución del vector de movimiento, etc.), derivar la información de
10 movimiento correspondiente.
- 2) Realizar la interpredicción para la plantilla de acuerdo con la información de movimiento derivada en la etapa (1) para generar la señal de predicción de la plantilla.
- 15 3) Obtener la señal de reconstrucción de la plantilla (muestras vecinas previamente decodificadas).
- 4) Calcular la distorsión entre la señal de reconstrucción y la señal de predicción.

Con respecto a la etapa 1 anterior, el
20 decodificador de video 300 puede, por ejemplo, configurarse para recibir elementos de sintaxis que indiquen un candidato AMVP y una MVD antes de realizar una adaptación de lista de imágenes de referencia a nivel de bloque. Por lo tanto, para cada hipótesis de referencia, el decodificador de video 300
25 puede determinar el vector de movimiento real que se usará para predecir el bloque y usar ese vector de movimiento para

determinar las hipótesis de referencia. Como diferentes imágenes de referencia pueden utilizar diferentes escalados de vectores de movimiento u otros procesos de derivación basados en imágenes de referencia, el vector de movimiento real utilizado para cada hipótesis de referencia puede ser diferente aunque el MVP y la MVD sean los mismos.

Usando, por ejemplo, las etapas 1-4 anteriores, el codificador de video 200 y el decodificador de video 300 pueden configurarse para generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte y generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte. Al usar dicho proceso de reordenamiento de imágenes de referencia adaptable a nivel de bloque, el codificador de video 200 y el decodificador de video 300 pueden reducir la sobrecarga de señalización asociada a índices de imagen de referencia de señalización en ciertos modos de codificación, tales como el modo AMVP.

En un ejemplo, el codificador de video 200 y el decodificador de video 300 pueden configurarse para reordenar las imágenes de referencia dentro de cada lista de referencia por separado. El diseño de sintaxis para imágenes de referencia puede ser el mismo que el de HEVC y VVC, con la diferencia de que el mismo valor de índice

de referencia puede indicar una imagen de referencia diferente para diferentes bloques.

En un ejemplo, el codificador de video 200 y el decodificador de video 300 pueden configurarse para reordenar las imágenes de referencia dentro de ambas listas de referencia juntas. Cuando se señala la dirección de interpredicción, en algunos ejemplos, solo se puede codificar un bin en el flujo de bits para indicar si un modo de predicción para un bloque es bipredicción o no predicción. En el caso de un modo de predicción única, un índice `ref_idx_uni` puede codificarse en el flujo de bits para indicar la imagen de referencia seleccionada a partir de una lista conjunta de imágenes de referencia que incluye imágenes de referencia tanto de la Lista 0 como de la Lista 1. Las imágenes de referencia en la lista conjunta se disponen de acuerdo con el orden ascendente del costo de la plantilla. En algunos ejemplos, puede que no sea necesario que haya un segundo bin para indicar la Lista 0 o la Lista 1, que en su lugar puede determinarse implícitamente por el índice. En el caso de un modo de bipredicción, un índice `ref_idx_bi` puede codificarse en el flujo de bits para indicar el par seleccionado de imágenes de referencia.

En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para reordenar las imágenes de referencia de ambas listas de referencia juntas para el modo de unipredicción, pero

reordenar, dentro de cada lista de referencia, por separado para el modo de bipredicción.

El codificador de video 200 y el decodificador de video 300 pueden configurarse para determinar un criterio de costos. De acuerdo con un ejemplo de esta divulgación, el costo es la distorsión entre la señal de predicción y la señal de reconstrucción de la plantilla. La medida de distorsión puede ser la suma de la diferencia absoluta (SAD, por sus siglas en inglés), la suma de la diferencia absoluta transformada (SATD, por sus siglas en inglés), la suma del error cuadrado (SSE, por sus siglas en inglés), u otra determinación de costos de este tipo.

En algunos ejemplos, el costo puede ser una combinación de distorsión entre la señal de predicción y la señal de reconstrucción de la plantilla y un término de regularización determinado por el índice de la hipótesis de imagen de referencia en la lista inicial. El término de regularización se puede derivar como el índice multiplicado por una escala. El valor de escala puede ser un entero constante predeterminado o puede ser un valor que depende del tamaño del bloque.

En otro ejemplo, cuando el modo de codificación es bipredicción, el codificador de video 200 y el decodificador de video 300 pueden configurarse para determinar el costo como la suma de distorsiones entre cada una de la señal de predicción y la señal de

reconstrucción. El término de regularización también se puede aplicar como se describió anteriormente.

El codificador de video 200 y el decodificador de video 300 pueden configurarse para realizar binarización del índice. En algunos ejemplos, se puede usar un código unario truncado para codificar el índice. En algunos ejemplos, se puede usar un código Golomb-Rice. El código Golomb-Rice es un proceso de codificación de longitud variable, formado con un código de prefijo unario truncado codificado y un código de sufijo codificado de longitud fija. Las longitudes respectivas de los códigos de prefijo y sufijo se controlan por un parámetro, divisor codificado D , donde D es una potencia positiva de 2 números (por ejemplo, 1, 2, 4, 8,...) y no es mayor que el número total de símbolos a codificar. Cuando hay N símbolos diferentes para codificarse usando código Golomb-Rice con un divisor D , el índice de estos N símbolos diferentes se descompone en grupos N/D y cada grupo es con símbolos D . El índice de grupo (por ejemplo, 0, 1, 2, 3,..., $N/D-1$) se codifica usando código unario truncado, que es el código prefijo antes mencionado. El índice (por ejemplo, 0, 1, 2, 3,..., $D-1$) de los símbolos D en un grupo se codifica entonces usando código binario truncado (cuando D no es potencia-de-2) o código binario de longitud fija (cuando D es un número de potencia de-2), que es el código sufijo antes mencionado. En algunos ejemplos, el código unario truncado se usa para `ref_idx_uni`, y el código Golomb-

Rice se usa para `ref_idx_bi`. El número de combinaciones de diferentes pares de imágenes de referencia en la bipredicción puede ser grande, por lo que se usa un gran número de valores de índice. Sin embargo, el código unario truncado puede no
5 ser eficiente para un valor de índice relativamente grande. En algunos ejemplos, el uso de código unario truncado o código Golomb-Rice depende del número total de valores de índice. Si el número es mayor que un umbral, entonces se usa el código Golomb-Rice; de lo contrario, se usa código unario
10 truncado.

La Figura 2 muestra un ejemplo de una plantilla. La plantilla generalmente corresponde o localiza el área codificada vecina inmediata de un bloque actual, tal como el bloque actual 140 en la Figura 2. Las muestras de la plantilla
15 se decodifican antes de que se decodificara el bloque actual.

En algunos ejemplos, la plantilla incluye una plantilla izquierda 142 a la izquierda del bloque actual 140 y/o una plantilla 144 anterior a la anterior del bloque actual 140, como se ilustra en la Figura 2. Si el ancho y
20 la altura del bloque actual 140 se indican como W y H , entonces el tamaño de la plantilla izquierda puede ser $N \times H$, y el tamaño del tamaño de plantilla anterior puede ser $W \times N$, donde N es un valor predefinido, como 1, 2, 4, o algún otro valor entero. También se pueden usar otras plantillas. Por
25 ejemplo, el tamaño de la plantilla izquierda puede ser $N \times 2H$, y el tamaño de plantilla de arriba puede ser $2W \times N$.

El codificador de video 200 y el decodificador de video 300 pueden configurarse para realizar la generación de señal de predicción para la plantilla. Como se describió anteriormente, el vector o vectores de movimiento se derivan primero de acuerdo con la hipótesis de imagen de referencia y otra información de movimiento relativo, y después se realiza la compensación de movimiento para generar la señal de predicción para la plantilla. El codificador de video 200 y el decodificador de video 300 pueden configurarse para usar el vector o vectores de movimiento determinados para localizar muestras en una imagen de referencia que corresponda a la plantilla. El codificador de video 200 y el decodificador de video 300 pueden determinar entonces el costo de emparejamiento de plantilla al comparar las muestras de plantilla de la imagen actual con las muestras correspondientes en la imagen de referencia.

En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para aplicar el filtro de interpolación usado para la compensación de movimiento de un bloque actual para la compensación de movimiento para la plantilla. En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden aplicar un filtro de interpolación simplificado para reducir la complejidad. Por ejemplo, se usa el filtro de interpolación bilineal en lugar del típico filtro de toque más largo (6 toques, 8 toques, 12 toques para ejemplos). En

algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para redondear el vector o vectores de movimiento al entero más cercano para la compensación de movimiento para la plantilla.

5 El codificador de video 200 y el decodificador de video 300 pueden configurarse para realizar la generación de una lista inicial. A continuación se describirá el caso de unipredicción. En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden
10 configurarse para generar la lista conjunta inicial de imágenes de referencia intercalando las imágenes de referencia de la Lista 0 y la Lista 1. La imagen de referencia j en la Lista i ($i = 0, 1$) se convierte en imagen de referencia $2*j+i$ en la lista conjunta cuando j es menor
15 que el número de imágenes de referencia en la Lista 0 y también es menor que el número de imágenes de referencia en la Lista 1. Cuando j es mayor o igual que el número de imágenes de referencia en la Lista 0 o es mayor o igual que el número de imágenes de referencia en la Lista 1, el índice
20 en la lista conjunta es $N + j$, en donde N es el mínimo entre el número de imágenes de referencia en la Lista 0 y el número de imágenes de referencia en la Lista 1.

 En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para
25 generar la lista conjunta inicial de imágenes de referencia al añadir primero las imágenes de referencia en la Lista

0 y después al añadir las imágenes de referencia en la
Lista 1. O, en algunos ejemplos, el codificador de video
200 y el decodificador de video 300 pueden configurarse
para generar la lista conjunta inicial de imágenes de
5 referencia al añadir primero las imágenes de referencia en
la Lista 1 y después al añadir las imágenes de referencia
en la Lista 0. En algunos ejemplos, el codificador de video
200 y el decodificador de video 300 pueden configurarse
para generar las imágenes de referencia en la lista
10 conjunta de acuerdo con el orden ascendente de distancia
entre la imagen de referencia y la imagen actual.

A continuación se describirá el caso de la
bipredicción. En un ejemplo, para la bipredicción, el
codificador de video 200 y el decodificador de video 300
15 pueden configurarse para generar la lista inicial de pares
de imágenes de referencia de la siguiente manera:

Haga un bucle sobre la imagen de referencia i en la
Lista 0

Haga un bucle sobre la imagen de referencia j en
20 la Lista 1

Inserte el par (i, j) en la lista.

La lista puede reordenarse según el orden
ascendente de $i+j$.

En otro ejemplo, el codificador de video 200 y el
25 decodificador de video 300 pueden configurarse para aplicar
un proceso de clasificación después de eso, donde se asigna

un costo a cada par de imágenes de referencia y la lista se ordena basada en el orden ascendente del costo. En un ejemplo, el costo se calcula como $\text{abs}(\text{pocDist0}) + \text{abs}(\text{pocDist1}) + \text{abs}(\text{pocDist0} - \text{pocDist1})$. pocDist0
5 representa la diferencia de un recuento de orden de imágenes (POC, por sus siglas en inglés) entre una imagen actual y la imagen de referencia 0. pocDist1 representa la diferencia de un POC entre la imagen de referencia 1 y una imagen actual, y $\text{abs}()$ es el operador para calcular el
10 valor absoluto. En otro ejemplo, el costo puede calcularse como $\text{abs}(\text{pocDist0}) + \text{abs}(\text{pocDist1}) + c$, en donde c se establece en 0 si pocDist0 and pocDist1 tienen signos diferentes y se establece en un valor positivo grande predefinido de cualquier otra manera. De esta manera, al
15 par de imágenes de referencia que tiene tanto imágenes de referencia hacia adelante como ambas referencias hacia atrás se le asigna un costo mayor.

El codificador de video 200 y el decodificador de video 300 pueden configurarse para manejar imágenes de
20 referencia duplicadas. Se puede insertar una misma imagen de referencia tanto en la Lista 0 como en la Lista 1. Al crear la lista conjunta para reordenar las imágenes de referencia para unipredicción, puede existir una imagen de referencia duplicada. En algunos ejemplos, se puede
25 asignar un gran costo predefinido a la imagen de referencia duplicada; por ejemplo, el costo se establece en el valor

máximo. En algunos ejemplos, el costo de plantilla para la imagen de referencia duplicada se escala mediante un valor predefinido, por ejemplo, 2, 4, 8, 16, etc. De manera similar, el mismo proceso puede aplicarse al par duplicado
5 de imágenes de referencia en el modo de bipredicción.

Se pueden usar diversas técnicas descritas en la presente descripción en combinación con predicción de signos para MVD. Los signos de MVD pueden predecirse basados en un costo de emparejamiento de plantilla, como se describe en
10 JVET-X0132 (24.^a Reunión, por teleconferencia, 6-15 de octubre de 2021) y JVET-K0067 (11.^a Reunión: Liubliana, SI, 10-18 de julio de 2018). En algunos ejemplos, el índice de imagen de referencia y los signos de MVD pueden codificarse juntos como un índice, en donde el valor del índice indica
15 una cierta combinación de hipótesis de imagen o imágenes de referencia e hipótesis de signos de MVD. Todas las combinaciones posibles se pueden pedir de acuerdo con el costo de emparejamiento de plantilla como se analizó anteriormente.

20 En algunos ejemplos, la predicción de signos para MVD y la adaptación de imágenes de referencia pueden ser mutuamente excluyentes. Las dos herramientas se pueden controlar mediante sintaxis de alto nivel, y solo una de las dos herramientas se puede aplicar a la imagen/corte
25 codificado actual.

En algunos ejemplos, la predicción de signos para MVD y la adaptación de imágenes de referencia son mutuamente excluyentes a nivel de bloque. Las dos herramientas pueden ser controladas por la magnitud de las MVD. Si la magnitud de MVD es inferior a algún umbral, se aplica la adaptación de imagen de referencia; de lo contrario, se aplica la predicción de signos para MVD.

En otro ejemplo, la predicción de signos para MVD y la adaptación de imágenes de referencia, como se analizó anteriormente, pueden usarse juntas. Cuando se realiza el reordenamiento de imágenes de referencia, el codificador de video 200 y el decodificador de video 300 pueden configurarse para asignar el costo mínimo entre todas las posibles hipótesis de MVD a la hipótesis de imagen o imágenes de referencia para el reordenamiento de imágenes de referencia. A continuación, dada la lista de imágenes de referencia reordenadas y el índice decodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar las imágenes de referencia seleccionadas, que pueden usarse posteriormente para la predicción de signos para las MVD. El costo calculado durante el proceso de reordenamiento de imágenes de referencia puede almacenarse y los asociados con las imágenes de referencia determinadas pueden reusarse para la predicción de signos.

En otro ejemplo, la predicción de signos para MVD y la adaptación de imagen de referencia propuesta,

como se analizó anteriormente, pueden usarse juntas. Cuando se realiza la predicción de signos para las MVD, el codificador de video 200 y el decodificador de video 300 pueden configurarse para asignar el costo mínimo entre
5 todas las posibles hipótesis de imágenes de referencia a la hipótesis de signos para las MVD. A continuación, dada la hipótesis de signo reordenado y el índice de signo decodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar las MVD
10 reales, que pueden usarse posteriormente para el reordenamiento de imágenes de referencia. El costo calculado durante el proceso de predicción de signos puede almacenarse y los asociados con las MVD determinadas pueden reusarse para el reordenamiento de imágenes de referencia.

15 En otro ejemplo, la predicción de signos para MVD y la adaptación de imágenes de referencia como se propuso anteriormente pueden usarse juntas. Cuando se realiza el reordenamiento de imágenes de referencia, el codificador de video 200 y el decodificador de video 300
20 pueden asignar el costo mínimo entre todas las posibles hipótesis de MVD a la hipótesis de imagen o imágenes de referencia para el reordenamiento de imágenes de referencia. Entonces, dada la lista de imágenes de referencia reordenadas y el índice decodificado, el
25 codificador de video 200 y el decodificador de video 300 pueden determinar las imágenes de referencia

seleccionadas. Al realizar la predicción de signos para las MVD, el codificador de video 200 y el decodificador de video 300 pueden asignar el costo mínimo entre todas las posibles hipótesis de imagen de referencia a la hipótesis de signos para las MVD. A continuación, dada la hipótesis de signo reordenado y el índice de signo decodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar las MVD reales.

Las técnicas descritas anteriormente pueden aplicarse de manera similar a una combinación de adaptación de imagen de referencia y otra predicción de sintaxis. La sintaxis puede ser, por ejemplo, un índice IMV, un indicador de compensación de iluminación local (LIC, por sus siglas en inglés), una bipredicción con índice de pesos a nivel de Cu (BCW, por sus siglas en inglés), o similares.

A continuación se describirán técnicas para la simplificación de la predicción de signos de MVD. Como se describió anteriormente, la combinación de reordenamiento de imágenes de referencia y predicción de signos de MVD puede implicar probar el costo de todas las combinaciones posibles de diferentes hipótesis, lo que podría imponer una complejidad computacional significativa para la bipredicción donde hay múltiples MVD y especialmente para el modo afín cuando hay como máximo tres MVD en cada dirección. Se pueden aplicar las siguientes técnicas para simplificar el proceso.

En un ejemplo, el codificador de video 200 y el decodificador de video 300 pueden configurarse para desactivar la predicción de signos de MVD para los modos de bipredicción. Se puede establecer una determinada
5 condición para la desactivación. Como condición de ejemplo, el modo no debe ser el modo SMVD para la desactivación, siendo simétricas las MVD de la Lista 0 y de la Lista 1. Como otra condición de ejemplo, el modo será el modo afín.

En otro ejemplo, el codificador de video 200 y
10 el decodificador de video 300 pueden configurarse para desactivar la predicción de signos de MVD para una de las direcciones en el caso de bipredicción. Por ejemplo, MVD en la Lista 1 puede desactivarse en el caso de bipredicción. Se puede establecer una determinada
15 condición para la desactivación. Por ejemplo, si MVD en la Lista 0 es distinto de cero, entonces la predicción del signo MVD para la Lista 1 está deshabilitada. Tenga en cuenta que las magnitudes de MVD en ambas direcciones (Lista 0 y Lista 1) se pueden codificar de antemano, y
20 después se puede determinar si la predicción de signos de MVD está deshabilitada para una de las listas.

En otro ejemplo más, el codificador de video 200 y el decodificador de video 300 pueden configurarse para limitar el número de signos de MVD a predecir. El número
25 puede ser configurable y señalizado en sintaxis de alto nivel. En un ejemplo, se puede establecer un contador para

los signos MVD, y una vez que el número de signos MVD predichos alcanza un número predefinido, entonces el resto de los signos MVD se codificarán regularmente, por ejemplo, use un bin para indicar signo negativo o positivo, en lugar de predicho. En otro ejemplo, las MVD se ordenan en función de las magnitudes en orden descendente. Después se predice hasta N MVD en la lista ordenada, y no en la restablecida. Tenga en cuenta que en ambos ejemplos, las magnitudes de las MVD tanto en la Lista 0 como en la Lista 1 pueden codificarse de antemano.

Las técnicas descritas anteriormente pueden aplicarse para la predicción de signos de MVD de forma independiente o pueden estar condicionadas por si se aplica una técnica de reordenamiento de referencia (u otra técnica de predicción de elementos de sintaxis).

El codificador de video 200 y el decodificador de video 300 pueden configurarse para determinar un costo de emparejamiento bilateral para el modo de bipredicción. En el caso del modo de bipredicción, se puede usar un costo de emparejamiento bilateral, en donde el costo es la distorsión entre el predictor de la Lista 0 y el predictor de la Lista 1. Un predictor de Lista 0 se obtiene por compensación de movimiento usando la información de movimiento derivada (de acuerdo con la hipótesis de la imagen de referencia) en la Lista 0, y un predictor de Lista 1 se obtiene por compensación de movimiento usando la

información de movimiento derivada (de acuerdo con la hipótesis de la imagen de referencia) en la Lista 1.

En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para determinar un costo de emparejamiento bilateral en lugar de un costo de emparejamiento de plantilla en el modo de bipredicción y después usar el costo de coincidencia bilateral para reordenar las listas de imágenes de referencia. En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden configurarse para usar el costo de emparejamiento bilateral junto con el costo de emparejamiento de plantilla en el modo de bipredicción. Por ejemplo, el codificador de video 200 y el decodificador de video 300 pueden configurarse para reordenar imágenes de referencia en función de una suma ponderada del costo de emparejamiento bilateral y el costo de emparejamiento de plantilla.

A continuación se describirán aspectos de la interacción con el remuestreo de imágenes de referencia. Si se aplica RPR a una imagen de referencia, normalmente se deshabilita el proceso de derivación del lado del decodificador. Por ejemplo, el decodificador de video 300 puede configurarse para deshabilitar el modo de emparejamiento de plantilla y la compensación de iluminancia lineal (LIC) si se aplica RPR a la imagen de referencia seleccionada. En consecuencia, el indicador LIC

(que indica el encendido/apagado de LIC) no se señala en el flujo de bits y se infiere que es falso (desactivado). Por otro lado, el índice MVP puede señalizarse en el flujo de bits en lugar de derivarse implícitamente mediante el emparejamiento de plantilla si se aplica RPR a la imagen de referencia seleccionada. Sin embargo, si se aplica el proceso de reordenamiento de referencia a nivel de bloque, la imagen de referencia seleccionada puede ser desconocida durante la etapa de análisis (ya que el reordenamiento de imágenes de referencia se realiza durante la etapa de construcción, que normalmente es mucho más lenta que el elemento de sintaxis de análisis y, por lo tanto, se realiza muy por detrás del análisis de los elementos de sintaxis relativos). Por lo tanto, si se aplica RPR a la imagen de referencia seleccionada puede ser desconocido al determinar si el indicador LIC (o índice MVP) está presente o no. Para resolver dichos problemas, a continuación se describen técnicas relacionadas con la interacción entre el reordenamiento de imágenes de referencia y el RPR.

En una técnica de ejemplo, el decodificador de video 300 puede configurarse de manera que el reordenamiento de imágenes de referencia y el RPR sean exclusivos. En un ejemplo, si el indicador de habilitación de RPR de nivel SPS es verdadero, entonces se infiere que el indicador de habilitación de reordenamiento de imágenes de referencia de nivel SPS es falso o se restringe que sea falso.

En otra técnica de ejemplo, si se escala una imagen de referencia o una de las imágenes de referencia en un par (por ejemplo, se aplica RPR), entonces el decodificador de video 300 se configura para establecer un costo correspondiente en el proceso de reordenamiento de la imagen de referencia o del par de imágenes de referencia a un valor máximo. De esta manera, las imágenes de referencia escaladas siempre pueden estar en la parte posterior de la lista reordenada, por ejemplo, una imagen de referencia escalada siempre tiene un índice mayor que el índice de una imagen de referencia no escalada. A nivel de corte o imagen, el número de imágenes de referencia no escaladas (o pares de imágenes de referencia) puede calcularse previamente, denotado como *NumNonScaled*. Después de analizar el índice de la imagen de referencia seleccionada en la lista reordenada, el decodificador de video 300 puede determinar si la imagen de referencia seleccionada es una imagen escalada comparando el valor de índice con N (sin realizar el proceso de reordenamiento). Por lo tanto, si el índice es mayor o igual que *NumNonScaled*, entonces la referencia seleccionada se puede determinar como una imagen de referencia escalada o determinar de cualquier otra manera que es no escalada.

La Figura 3 es un diagrama de bloques que ilustra un codificador de video 200 de ejemplo que puede realizar las técnicas de esta descripción. La Figura 3 se proporciona con fines de explicación y no debe considerarse limitante

de las técnicas como se ejemplifica y describe ampliamente en esta descripción. Para fines de explicación, esta descripción describe el codificador de video 200 de acuerdo con las técnicas de VVC (ITU-T H.266, en desarrollo) y HEVC (ITU-T H.265). Sin embargo, las técnicas de esta descripción pueden realizarse por dispositivos de codificación de video que se configuran para otros estándares de codificación de video y formatos de codificación de video, tales como AV1 y sucesores del formato de codificación de video AV1.

En el ejemplo de la Figura 3, el codificador de video 200 incluye la memoria de datos de video 230, la unidad de selección de modo 202, la unidad de generación residual 204, la unidad de procesamiento de transformada 206, la unidad de cuantificación 208, la unidad de cuantificación inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214, la unidad de filtro 216, el búfer de imágenes decodificadas (DPB) 218 y la unidad de codificación por entropía 220. Cualquiera o la totalidad de la memoria de datos de video 230, la unidad de selección de modo 202, la unidad de generación residual 204, la unidad de procesamiento de transformada 206, la unidad de cuantificación 208, la unidad de cuantificación inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214, la unidad de filtro 216, DPB 218 y la unidad de codificación por entropía 220 pueden

implementarse en uno o más procesadores o en circuitos de procesamiento. Por ejemplo, las unidades del codificador de video 200 pueden implementarse como uno o más circuitos o elementos lógicos como parte de circuitos de *hardware*, o
5 como parte de un procesador, ASIC o FPGA. Además, el codificador de video 200 puede incluir procesadores o circuitos de procesamiento adicionales o alternativos para realizar estas y otras funciones.

La memoria de datos de video 230 puede almacenar
10 datos de video para codificarse por los componentes del codificador de video 200. El codificador de video 200 puede recibir los datos de video almacenados en la memoria de datos de video 230 desde, por ejemplo, la fuente de video 104 (Figura 1). El DPB 218 puede actuar como una memoria
15 de imagen de referencia que almacena datos de video de referencia para el uso en la predicción de datos de video subsiguientes por el codificador de video 200. La memoria de datos de video 230 y el DPB 218 pueden estar formadas por cualquiera de una variedad de dispositivos de memoria,
20 tales como memoria de acceso aleatorio dinámico (DRAM, por sus siglas en inglés), incluidas DRAM síncrona (SDRAM, por sus siglas en inglés), RAM magnetorresistiva (MRAM, por sus siglas en inglés), RAM resistiva (RRAM, por sus siglas en inglés) u otros tipos de dispositivos de memoria. La
25 memoria de datos de video 230 y DPB 218 pueden proporcionarse mediante el mismo dispositivo de memoria o

dispositivos de memoria separados. En diversos ejemplos, la memoria de datos de video 230 puede estar en chip con otros componentes del codificador de video 200, como se ilustra, o fuera del chip en relación con esos componentes.

5 En esta divulgación, la referencia a la memoria de datos de video 230 no debe interpretarse como limitada a la memoria interna del codificador de video 200, a menos que se describan específicamente como tal, o a la memoria externa al codificador de video 200, a menos que se
10 describan específicamente como tal. Más bien, la referencia a la memoria de datos de video 230 debe entenderse como una memoria de referencia que almacena datos de video que el codificador de video 200 recibe para su codificación (por ejemplo, datos de video para un bloque actual que se
15 codificará). La memoria 106 de la Figura 1 también puede proporcionar almacenamiento temporal de salidas de las diversas unidades del codificador de video 200.

Las diversas unidades de la Figura 3 se ilustran para ayudar a comprender las operaciones realizadas por el
20 codificador de video 200. Las unidades pueden implementarse como circuitos de función fija, circuitos programables, o una combinación de estos. Los circuitos de función fija se refieren a circuitos que proporcionan una funcionalidad particular y están preestablecidos en las operaciones que
25 se pueden realizar. Los circuitos programables se refieren a circuitos que se pueden programar para realizar diversas

tareas, y proporcionan una funcionalidad flexible en las operaciones que se pueden realizar. Por ejemplo, los circuitos programables pueden ejecutar *software* o *firmware* que hacen que los circuitos programables funcionen de la manera definida por las instrucciones del *software* o *firmware*. Los circuitos de función fija pueden ejecutar instrucciones de *software* (por ejemplo, para recibir parámetros o parámetros de salida), pero los tipos de operaciones que realizan los circuitos de función fija son generalmente inmutables. En algunos ejemplos, una o más de las unidades pueden ser bloques de circuito distintos (función fija o programable), y en algunos ejemplos, una o más de las unidades pueden ser circuitos integrados.

El codificador de video 200 puede incluir unidades lógicas aritméticas (ALU, por sus siglas en inglés), unidades de función elemental (EFU, por sus siglas en inglés), circuitos digitales, circuitos analógicos y/o núcleos programables, formados a partir de circuitos programables. En ejemplos en los que las operaciones del codificador de video 200 se realizan usando *software* ejecutado por los circuitos programables, la memoria 106 (Figura 1) puede almacenar las instrucciones (por ejemplo, código objeto) del *software* que recibe y ejecuta el codificador de video 200, u otra memoria dentro del codificador de video 200 (no mostrado) puede almacenar dichas instrucciones.

La memoria de datos de video 230 se configura para almacenar datos de video recibidos. El codificador de video 200 puede recuperar una imagen de los datos de video de la memoria de datos de video 230 y proporcionar los
5 datos de video a la unidad de generación residual 204 y a la unidad de selección de modo 202. Los datos de video en la memoria de datos de video 230 pueden ser datos de video sin procesar que se van a codificar.

La unidad de selección de modo 202 incluye una
10 unidad de estimación de movimiento 222, una unidad de compensación de movimiento 224 y una unidad de intrapredicción 226. La unidad de selección de modo 202 puede incluir unidades funcionales adicionales para realizar predicción de video de acuerdo con otros modos de
15 predicción. Como ejemplos, la unidad de selección de modo 202 puede incluir una unidad de paleta, una unidad de copia intrabloque (que puede ser parte de la unidad de estimación de movimiento 222 y/o la unidad de compensación de movimiento 224), una unidad afín, una unidad de modelo
20 lineal (LM, por sus siglas en inglés), o similares.

La unidad de selección de modo 202 coordina generalmente múltiples pasadas de codificación para probar combinaciones de parámetros de codificación y valores de distorsión de tasa resultantes para dichas combinaciones.
25 Los parámetros de codificación pueden incluir la partición de las CTU en CU, modos de predicción para las CU, tipos de

transformada para datos residuales de las CU, parámetros de cuantificación para datos residuales de las CU, etc. La unidad de selección de modo 202 puede seleccionar en última instancia la combinación de parámetros de codificación que
5 tienen valores de distorsión de tasa que son mejores que las otras combinaciones probadas.

El codificador de video 200 puede particionar una imagen recuperada de la memoria de datos de video 230 en una serie de CTU, y encapsular una o más CTU dentro de
10 un corte. La unidad de selección de modo 202 puede particionar una CTU de la imagen de acuerdo con una estructura de árbol, tal como la estructura MTT, estructura QTBT, estructura de superbloque, o la estructura de árbol cuádruple descrita anteriormente. Como se describió
15 anteriormente, el codificador de video 200 puede formar una o más CU a partir de la partición de una CTU de acuerdo con la estructura de árbol. Tal CU también puede denominarse generalmente “bloque de video” o “bloque”.

En general, la unidad de selección de modo 202
20 también controla los componentes de esta (por ejemplo, la unidad de estimación de movimiento 222, la unidad de compensación de movimiento 224 y la unidad de intrapredicción 226) para generar un bloque de predicción para un bloque actual (por ejemplo, una CU actual, o en
25 HEVC, la porción superpuesta de una PU y una TU). Para la interpredicción de un bloque actual, la unidad de

estimación de movimiento 222 puede realizar una búsqueda de movimiento para identificar uno o más bloques de referencia estrechamente coincidentes en una o más imágenes de referencia (por ejemplo, una o más imágenes codificadas previamente almacenadas en DPB 218). En particular, la unidad de estimación de movimiento 222 puede calcular un valor representativo de cuán similar es un bloque de referencia potencial al bloque actual, por ejemplo, de acuerdo con la suma de la diferencia absoluta (SAD), la suma de diferencias cuadradas (SSD), la diferencia absoluta media (MAD), las diferencias cuadradas medias (MSD), o similares. La unidad de estimación de movimiento 222 generalmente puede realizar estos cálculos usando diferencias muestra por muestra entre el bloque actual y el bloque de referencia que se está considerando. La unidad de estimación de movimiento 222 puede identificar un bloque de referencia que tiene un valor más bajo resultante de estos cálculos, indicando un bloque de referencia que coincide más estrechamente con el bloque actual.

La unidad de estimación de movimiento 222 puede formar uno o más vectores de movimiento (MV, por sus siglas en inglés) que definen las posiciones de los bloques de referencia en las imágenes de referencia en relación con la posición del bloque actual en una imagen actual. La unidad de estimación de movimiento 222 puede proporcionar entonces los vectores de movimiento a la unidad de

compensación de movimiento 224. Por ejemplo, para la
interpredicción unidireccional, la unidad de estimación de
movimiento 222 puede proporcionar un único vector de
movimiento, mientras que para la interpredicción
5 bidireccional, la unidad de estimación de movimiento 222
puede proporcionar dos vectores de movimiento. La unidad
de compensación de movimiento 224 puede generar entonces
un bloque de predicción usando los vectores de movimiento.
Por ejemplo, la unidad de compensación de movimiento 224
10 puede recuperar datos del bloque de referencia usando el
vector de movimiento. Como otro ejemplo, si el vector de
movimiento tiene una precisión de muestra fraccional, la
unidad de compensación de movimiento 224 puede interpolar
valores para el bloque de predicción de acuerdo con uno o
15 más filtros de interpolación. Además, para la
interpredicción bidireccional, la unidad de compensación
de movimiento 224 puede recuperar datos para dos bloques
de referencia identificados por vectores de movimiento
respectivos y combinar los datos recuperados, por ejemplo,
20 a través del promedio muestra por muestra o el promedio
ponderado.

Cuando se opera de acuerdo con el formato de
codificación de video AV1, la unidad de estimación de
movimiento 222 y la unidad de compensación de movimiento
25 224 pueden configurarse para codificar bloques de
codificación de datos de video (por ejemplo, bloques de

codificación de luma y croma) usando compensación de movimiento traslacional, compensación de movimiento afín, compensación de movimiento en bloque superpuesto (OBMC) y/o predicción inter-intracompuesta.

5 Como parte de determinar cómo codificar un bloque y después como parte de descodificar posteriormente el bloque codificado, la unidad de estimación de movimiento 222 y la unidad de compensación de movimiento 224 pueden configurarse para realizar diversas técnicas de esta descripción. Por ejemplo, la unidad de estimación de movimiento 222 y/o la unidad de compensación de movimiento 224 pueden configurarse para generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte 10 y después generar un segundo ordenamiento diferente de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte basado en un proceso de reordenamiento adaptable de imágenes de referencia. La unidad de compensación de movimiento 224 15 puede configurarse para decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia y decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia. 20

 Como otro ejemplo, para la intrapredicción, o 25 codificación intrapredicción, la unidad de intrapredicción 226 puede generar el bloque de predicción a partir de

muestras vecinas al bloque actual. Por ejemplo, para modos direccionales, la unidad de intrapredicción 226 puede generalmente combinar matemáticamente valores de muestras vecinas y rellenar estos valores calculados en la dirección definida a través del bloque actual para producir el bloque de predicción. Como otro ejemplo, para el modo DC, la unidad de intrapredicción 226 puede calcular un promedio de las muestras vecinas al bloque actual y generar el bloque de predicción para incluir este promedio resultante para cada muestra del bloque de predicción.

Cuando se opera de acuerdo con el formato de codificación de video AV1, la unidad de intrapredicción 226 puede configurarse para codificar bloques de codificación de datos de video (por ejemplo, bloques de codificación tanto de luma como de croma) usando predicción intra direccional, intrapredicción no direccional, intra predicción de filtro recursivo, predicción croma-de-luma (CFL, por sus siglas en inglés), copia intrabloque (IBC, por sus siglas en inglés) y/o modo de paleta de colores. La unidad de selección de modo 202 puede incluir unidades funcionales adicionales para realizar predicción de video de acuerdo con otros modos de predicción.

La unidad de selección de modo 202 proporciona el bloque de predicción a la unidad de generación residual 204. La unidad de generación residual 204 recibe una versión sin codificar del bloque actual desde la memoria de datos de

video 230 y el bloque de predicción de la unidad de selección de modo 202. La unidad de generación residual 204 calcula las diferencias muestra por muestra entre el bloque actual y el bloque de predicción. Las diferencias resultantes muestra por muestra definen un bloque residual para el bloque actual. En algunos ejemplos, la unidad de generación residual 204 también puede determinar diferencias entre los valores de muestra en el bloque residual para generar un bloque residual usando modulación de código de pulso diferencial residual (RDPCM, por sus siglas en inglés). En algunos ejemplos, la unidad de generación residual 204 puede formarse usando uno o más circuitos sustractores que realizan sustracción binaria.

En ejemplos donde la unidad de selección de modo 202 divide las CU en PU, cada PU puede estar asociada con una unidad de predicción de luma y unidades de predicción de croma correspondientes. El codificador de video 200 y el decodificador de video 300 pueden soportar PU que tienen varios tamaños. Como se indicó anteriormente, el tamaño de una CU puede referirse al tamaño del bloque de codificación de luma de la CU y el tamaño de una PU puede referirse al tamaño de una unidad de predicción de luma de la PU. Suponiendo que el tamaño de una CU particular es $2N \times 2N$, el codificador de video 200 puede soportar tamaños de PU de $2N \times 2N$ o $N \times N$ para la intrapredicción, y tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ o similares para la

interpredicción. El codificador de video 200 y el decodificador de video 300 también pueden soportar particiones asimétricas para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$ para la interpredicción.

5 En ejemplos en los que la unidad de selección de modo 202 no divide adicionalmente una CU en PU, cada CU puede estar asociada con un bloque de codificación de luma y bloques de codificación de croma correspondientes. Como anteriormente, el tamaño de una CU puede referirse al
10 tamaño del bloque de codificación de luma de la CU. El codificador de video 200 y el decodificador de video 300 pueden soportar tamaños de CU de $2N \times 2N$, $2N \times N$ o $N \times 2N$.

Para otras técnicas de codificación de video tales como una codificación en modo de copia intrabloque,
15 una codificación en modo afín y codificación en modo de modelo lineal (LM), como algunos ejemplos, la unidad de selección de modo 202, a través de unidades respectivas asociadas con las técnicas de codificación, genera un bloque de predicción para el bloque actual que se está
20 codificando. En algunos ejemplos, como la codificación en modo paleta, la unidad de selección de modo 202 puede no generar un bloque de predicción, y en su lugar generar elementos de sintaxis que indican la manera en la que se reconstruye el bloque en función de una paleta
25 seleccionada. En tales modos, la unidad de selección de

modo 202 puede proporcionar estos elementos de sintaxis a la unidad de codificación por entropía 220 a ser codificada.

Como se describió anteriormente, la unidad de generación residual 204 recibe los datos de video para el bloque actual y el bloque de predicción correspondiente. La unidad de generación residual 204 genera entonces un bloque residual para el bloque actual. Para generar el bloque residual, la unidad de generación residual 204 calcula las diferencias muestra por muestra entre el bloque de predicción y el bloque actual.

La unidad de procesamiento de transformada 206 aplica una o más transformadas al bloque residual para generar un bloque de coeficientes de transformada (denominado en la presente descripción “bloque de coeficiente de transformada”). La unidad de procesamiento de transformada 206 puede aplicar diversas transformadas a un bloque residual para formar el bloque de coeficiente de transformada. Por ejemplo, la unidad de procesamiento de transformada 206 puede aplicar una transformada de coseno discreta (DCT), una transformada direccional, una transformada de Karhunen-Loeve (KLT) o una transformada conceptualmente similar a un bloque residual. En algunos ejemplos, la unidad de procesamiento de transformada 206 puede realizar múltiples transformadas a un bloque residual, por ejemplo, una transformada primaria y una transformada secundaria, tal como una transformada

rotacional. En algunos ejemplos, la unidad de procesamiento de transformada 206 no aplica transformadas a un bloque residual.

Cuando se opera de acuerdo con AV1, la unidad de procesamiento de transformada 206 puede aplicar una o más transformada al bloque residual para generar un bloque de coeficientes de transformada (denominado en la presente descripción “bloque de coeficiente de transformada”). La unidad de procesamiento de transformada 206 puede aplicar diversas transformadas a un bloque residual para formar el bloque de coeficiente de transformada. Por ejemplo, la unidad de procesamiento de transformada 206 puede aplicar una combinación de transformada horizontal/vertical que puede incluir una transformada de coseno discreta (DCT), una transformada de seno discreta asimétrica (ADST, por sus siglas en inglés), una ADST volteada (por ejemplo, una ADST en orden inverso) o una transformada de identidad (IDTX, por sus siglas en inglés). Cuando se usa una transformada de identidad, la transformada se omite en una de las direcciones vertical u horizontal. En algunos ejemplos, se puede omitir el procesamiento de transformada.

La unidad de cuantificación 208 puede cuantificar los coeficientes de transformada en un bloque de coeficientes de transformada, para producir un bloque de coeficiente de transformada cuantificado. La unidad de

cuantificación 208 puede cuantificar coeficientes de transformada de un bloque de coeficientes de transformada de acuerdo con un valor de parámetro de cuantificación (QP, por sus siglas en inglés) asociado con el bloque actual. El codificador de video 200 (por ejemplo, a través de la unidad de selección de modo 202) puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes de transformada asociados con el bloque actual al ajustar el valor de QP asociado con la CU. La cuantificación puede introducir pérdida de información y, por lo tanto, los coeficientes de transformada cuantificados pueden tener menor precisión que los coeficientes de transformada originales producidos por la unidad de procesamiento de transformada 206.

La unidad de cuantificación inversa 210 y la unidad de procesamiento de transformada inversa 212 pueden aplicar cuantificación inversa y transformada transformadas inversas a un bloque de coeficientes de transformada cuantificado, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficientes de transformada. La unidad de reconstrucción 214 puede producir un bloque reconstruido correspondiente al bloque actual (aunque potencialmente con algún grado de distorsión) basado en el bloque residual reconstruido y un bloque de predicción generado por la unidad de selección de modo 202. Por ejemplo, la unidad de reconstrucción 214 puede añadir muestras del

bloque residual reconstruido a muestras correspondientes del bloque de predicción generado por la unidad de selección de modo 202 para producir el bloque reconstruido.

La unidad de filtro 216 puede realizar una o más
5 operaciones de filtro en bloques reconstruidos. Por ejemplo, la unidad de filtro 216 puede realizar operaciones de desbloqueo para reducir los artefactos de fragmentación en bloques a lo largo de los bordes de las Cu. Las operaciones de la unidad de filtro 216 pueden omitirse, en
10 algunos ejemplos.

Cuando se opera de acuerdo con AV1, la unidad de filtro 216 puede realizar una o más operaciones de filtro en bloques reconstruidos. Por ejemplo, la unidad de filtro 216 puede realizar operaciones de desbloqueo para reducir
15 los artefactos de fragmentación en bloques a lo largo de los bordes de las Cu. En otros ejemplos, la unidad de filtro 216 puede aplicar un filtro de mejora direccional restringido (CDEF, por sus siglas en inglés), que puede aplicarse después del desbloqueo, y puede incluir la
20 aplicación de filtros direccionales de paso bajo no separables, no lineales, basados en direcciones de borde estimadas. La unidad de filtro 216 también puede incluir un filtro de restauración de bucle, que se aplica después de CDEF, y puede incluir un filtro Wiener normalizado
25 simétrico separable o un filtro autoguiado dual.

El codificador de video 200 almacena bloques reconstruidos en DPB 218. Por ejemplo, en ejemplos en los que no se realizan operaciones de la unidad de filtro 216, la unidad de reconstrucción 214 puede almacenar bloques reconstruidos en DPB 218. En ejemplos donde se realizan operaciones de la unidad de filtro 216, la unidad de filtro 216 puede almacenar los bloques reconstruidos filtrados en DPB 218. La unidad de estimación de movimiento 222 y la unidad de compensación de movimiento 224 pueden recuperar una imagen de referencia de DPB 218, formada a partir de los bloques reconstruidos (y potencialmente filtrados), para interpretar bloques de imágenes codificadas posteriormente. Además, la unidad de intrapredicción 226 puede usar bloques reconstruidos en DPB 218 de una imagen actual para intraprededir otros bloques en la imagen actual.

En general, la unidad de codificación por entropía 220 puede codificar por entropía elementos de sintaxis recibidos de otros componentes funcionales del codificador de video 200. Por ejemplo, la unidad de codificación por entropía 220 puede codificar por entropía bloques de coeficientes de transformada cuantificados a partir de la unidad de cuantificación 208. Como otro ejemplo, la unidad de codificación por entropía 220 puede codificar por entropía elementos de sintaxis de predicción (por ejemplo, información de movimiento para la interpredicción o información intramodo para

intrapredicción) a partir de la unidad de selección de modo 202. La unidad de codificación por entropía 220 puede realizar una o más operaciones de codificación de entropía en los elementos de sintaxis, que son otro ejemplo de datos
5 de video, para generar datos codificados por entropía. Por ejemplo, la unidad de codificación por entropía 220 puede realizar una operación de codificación de longitud variable adaptativa al contexto (CAVLC, por sus siglas en inglés), una operación CABAC, una operación de codificación de
10 longitud variable a variable (V2V, por sus siglas en inglés), una operación de codificación aritmética binaria adaptativa al contexto basada en sintaxis (SBAC, por sus siglas en inglés), una operación de codificación de entropía de partición de intervalo de probabilidad (PIPE,
15 por sus siglas en inglés), una operación de codificación exponencial-Golomb u otro tipo de entropía operación de codificación en los datos. En algunos ejemplos, la unidad de codificación por entropía 220 puede operar en modo de derivación donde los elementos de sintaxis no se codifican
20 por entropía.

El codificador de video 200 puede emitir un flujo de bits que incluye los elementos de sintaxis codificados por entropía necesarios para reconstruir bloques de un corte o imagen. En particular, la unidad de codificación
25 por entropía 220 puede emitir el flujo de bits.

De acuerdo con AV1, la unidad de codificación por entropía 220 puede configurarse como un codificador aritmético multisímbolo adaptativo de símbolo a símbolo. Un elemento de sintaxis en AV1 incluye un alfabeto de N
5 elementos, y un contexto (por ejemplo, modelo de probabilidad) incluye un conjunto de N probabilidades. La unidad de codificación por entropía 220 puede almacenar las probabilidades como funciones de distribución acumulativa (CDF, por sus siglas en inglés) de n bits (por
10 ejemplo, 15 bits). La unidad de codificación por entropía 22 puede realizar escalado recursivo, con un factor de actualización basado en el tamaño del alfabeto, para actualizar los contextos.

Las operaciones descritas anteriormente se
15 describen con respecto a un bloque. Dicha descripción debe entenderse como operaciones para un bloque de codificación de luma y/o bloques de codificación de croma. Como se describió anteriormente, en algunos ejemplos, el bloque de codificación de luma y los bloques de codificación de croma
20 son componentes de luma y croma de una CU. En algunos ejemplos, el bloque de codificación de luma y los bloques de codificación de croma son componentes de luma y croma de una PU.

En algunos ejemplos, las operaciones realizadas
25 con respecto a un bloque de codificación de luma no necesitan repetirse para los bloques de codificación de croma. Como

ejemplo, las operaciones para identificar un vector de movimiento (MV) y una imagen de referencia para un bloque de codificación de luma no necesitan repetirse para identificar un MV y una imagen de referencia para los bloques de cromina. Más bien, el MV para el bloque de codificación de luma puede escalarse para determinar el MV para los bloques de cromina, y la imagen de referencia puede ser la misma. Como otro ejemplo, el proceso de intrapredicción puede ser el mismo para el bloque de codificación de luma y los bloques de codificación de cromina.

El codificador de video 200 representa un ejemplo de un dispositivo configurado para codificar datos de video que incluye una memoria configurada para almacenar datos de video y una o más unidades de procesamiento implementadas en circuitos y configuradas para generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque y generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque, en donde el primer ordenamiento es diferente del segundo ordenamiento.

La Figura 4 es un diagrama de bloques que ilustra un decodificador de video 300 de ejemplo que puede realizar las técnicas de esta divulgación. La Figura 4 se proporciona con fines de explicación y no limita las técnicas ejemplificadas y descritas ampliamente en esta divulgación. Para fines de explicación, esta divulgación describe el

decodificador de video 300 según las técnicas de VVC (ITU-T H.266, en desarrollo), y HEVC (ITU-T H.265). Sin embargo, las técnicas de esta divulgación pueden ser realizadas por dispositivos de codificación de video configurados para
5 otros estándares de codificación de video.

En el ejemplo de la Figura 4, el decodificador de video 300 incluye la memoria de búfer de imágenes codificadas (CPB, por sus siglas en inglés) 320, la unidad de decodificación por entropía 302, la unidad de
10 procesamiento de predicción 304, la unidad de cuantificación inversa 306, la unidad de procesamiento de transformada inversa 308, la unidad de reconstrucción 310, la unidad de filtro 312 y el búfer de imágenes decodificadas (DPB, por sus siglas en inglés) 314. Cualquiera o todos de
15 la memoria de CPB 320, la unidad de decodificación por entropía 302, la unidad de procesamiento de predicción 304, la unidad de cuantificación inversa 306, la unidad de procesamiento de transformada inversa 308, la unidad de reconstrucción 310, la unidad de filtro 312 y el DPB 314
20 pueden implementarse en uno o más procesadores o en circuitos de procesamiento. Por ejemplo, las unidades del decodificador de video 300 pueden implementarse como uno o más circuitos o elementos lógicos como parte de un circuito de *hardware*, o como parte de un procesador, ASIC o FPGA.
25 Además, el decodificador de video 300 puede incluir

procesadores o circuitos de procesamiento adicionales o alternativos para realizar estas y otras funciones.

La unidad de procesamiento de predicción 304 incluye la unidad de compensación de movimiento 316 y la
5 unidad de intrapredicción 318. La unidad de procesamiento de predicción 304 puede incluir unidades adicionales para realizar la predicción de acuerdo con otros modos de predicción. Como ejemplos, la unidad de procesamiento de predicción 304 puede incluir una unidad de paleta, una
10 unidad de copia intrabloque (que puede formar parte de la unidad de compensación de movimiento 316), una unidad afín, una unidad de modelo lineal (LM, por sus siglas en inglés), o similares. En otros ejemplos, el decodificador de video 300 puede incluir más, menos o diferentes componentes
15 funcionales.

Cuando se opera de acuerdo con AV1, la unidad de compensación de movimiento 316 puede configurarse para decodificar bloques de codificación de datos de video (por ejemplo, bloques de codificación de luma y croma) usando
20 compensación de movimiento traslacional, compensación de movimiento afín, OBMC y/o inter-intrapredicción compuesta, como se describió anteriormente. La unidad de intrapredicción 318 puede configurarse para decodificar bloques de codificación de datos de video (por ejemplo,
25 bloques de codificación tanto de luma como de croma) usando predicción intradireccional, intrapredicción no

direccional, intrapredicción de filtro recursivo, CFL, copia intrabloque (IBC) y/o modo de paleta de colores, como se describió anteriormente.

La unidad de compensación de movimiento 316 puede configurarse para realizar diversas técnicas de esta descripción. Por ejemplo, la unidad de compensación de movimiento 316 puede configurarse para generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte y después generar un segundo ordenamiento diferente de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte basado en un proceso de reordenamiento adaptable de imágenes de referencia. La unidad de compensación de movimiento 316 puede configurarse para decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia y decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia.

La memoria de CPB 320 puede almacenar datos de video, tales como un flujo de bits de video codificado, a ser decodificado por los componentes del decodificador de video 300. Los datos de video almacenados en la memoria de CPB 320 pueden obtenerse, por ejemplo, a partir del medio legible por computador 110 (figura 1). La memoria de CPB 320 puede incluir un CPB que almacena datos de video codificados (por ejemplo, elementos de sintaxis) de un flujo de bits de video codificado. Además, la memoria de

CPB 320 puede almacenar datos de video distintos de los elementos de sintaxis de una imagen codificada, tales como datos temporales que representan salidas de las diversas unidades del decodificador de video 300. El DPB 314
5 generalmente almacena imágenes decodificadas, que el decodificador de video 300 puede emitir y/o usar como datos de video de referencia cuando decodifica datos o imágenes subsiguientes del flujo de bits de video codificado. La memoria de CPB 320 y el DPB 314 pueden estar formados por
10 cualquiera de una variedad de dispositivos de memoria, tales como DRAM, incluidos SDRAM, MRAM, RRAM u otros tipos de dispositivos de memoria. La memoria de CPB 320 y el DPB 314 pueden ser proporcionados por el mismo dispositivo de memoria o dispositivos de memoria separados. En diversos
15 ejemplos, la memoria de CPB 320 puede estar en chip con otros componentes del decodificador de video 300, o fuera del chip en relación con esos componentes.

Adicional o alternativamente, en algunos ejemplos, el decodificador de video 300 puede recuperar datos de video
20 codificados de la memoria 120 (Figura 1). Es decir, la memoria 120 puede almacenar datos como los analizados anteriormente con la memoria de CPB 320. De la misma manera, la memoria 120 puede almacenar instrucciones que han de ser ejecutadas por el decodificador de video 300, cuando parte o la totalidad
25 de la funcionalidad del decodificador de video 300 se

implementa en *software* para su ejecución por circuitos de procesamiento del decodificador de video 300.

Las diversas unidades mostradas en la Figura 4 se ilustran para ayudar a comprender las operaciones realizadas por el decodificador de video 300. Las unidades pueden implementarse como circuitos de función fija, circuitos programables, o una combinación de estos. De manera similar a la Figura 3, los circuitos de función fija se refieren a circuitos que proporcionan una funcionalidad particular, y están preestablecidos en las operaciones que se pueden realizar. Los circuitos programables se refieren a circuitos que se pueden programar para realizar diversas tareas, y proporcionan una funcionalidad flexible en las operaciones que se pueden realizar. Por ejemplo, los circuitos programables pueden ejecutar *software* o *firmware* que hacen que los circuitos programables funcionen de la manera definida por las instrucciones del *software* o *firmware*. Los circuitos de función fija pueden ejecutar instrucciones de *software* (por ejemplo, para recibir parámetros o parámetros de salida), pero los tipos de operaciones que realizan los circuitos de función fija son generalmente inmutables. En algunos ejemplos, una o más de las unidades pueden ser bloques de circuito distintos (función fija o programable), y en algunos ejemplos, una o más de las unidades pueden ser circuitos integrados.

El decodificador de video 300 puede incluir ALU, EFU, circuitos digitales, circuitos analógicos y/o núcleos programables formados a partir de circuitos programables. En ejemplos en los que las operaciones del decodificador de video 300 son realizadas por *software* que se ejecuta en los circuitos programables, la memoria en chip o fuera del chip puede almacenar instrucciones (por ejemplo, código objeto) del *software* que el decodificador de video 300 recibe y ejecuta.

La unidad de decodificación por entropía 302 puede recibir datos de video codificados desde el CPB y decodificar por entropía los datos de video para reproducir elementos de sintaxis. La unidad de procesamiento de predicción 304, la unidad de cuantificación inversa 306, la unidad de procesamiento de transformada inversa 308, la unidad de reconstrucción 310 y la unidad de filtro 312 pueden generar datos de video decodificados basados en los elementos de sintaxis extraídos del flujo de bits.

En general, el decodificador de video 300 reconstruye una imagen bloque por bloque. El decodificador de video 300 puede realizar una operación de reconstrucción en cada bloque individualmente (donde el bloque que se está reconstruyendo, es decir, decodificando, actualmente puede denominarse “bloque actual”).

La unidad de decodificación por entropía 302 puede decodificar por entropía elementos de sintaxis que definen

coeficientes de transformada cuantificados de un bloque de coeficientes de transformada cuantificados, así como información de transformada, tal como un parámetro de cuantificación (QP) y/o indicaciones de modo de transformada. La unidad de cuantificación inversa 306 puede usar el QP asociado con el bloque de coeficientes de transformada cuantificados para determinar un grado de cuantificación y, de la misma manera, un grado de cuantificación inversa para aplicar la unidad de cuantificación inversa 306. La unidad de cuantificación inversa 306 puede, por ejemplo, realizar una operación de desplazamiento a la izquierda bit a bit para cuantificar de manera inversa los coeficientes de transformada cuantificados. La unidad de cuantificación inversa 306 puede formar de esta manera un bloque de coeficientes de transformada que incluye coeficientes de transformada.

Después de que la unidad de cuantificación inversa 306 forme el bloque de coeficiente de transformada, la unidad de procesamiento de transformada inversa 308 puede aplicar una o más transformadas inversas al bloque de coeficientes de transformada para generar un bloque residual asociado con el bloque actual. Por ejemplo, la unidad de procesamiento de transformada inversa 308 puede aplicar una DCT inversa, una transformada entera inversa, una transformadatransformada inversa de Karhunen-Loeve (KLT) inversa, una transformada rotacional inversa, una

transformada direccional inversa u otra transformada transformada inversa al bloque de coeficientes de transformada.

Además, la unidad de procesamiento de predicción 304 genera un bloque de predicción de acuerdo con elementos de sintaxis de información de predicción que fueron decodificados por entropía por la unidad de decodificación por entropía 302. Por ejemplo, si los elementos de sintaxis de información de predicción indican que el bloque actual está interpredicho, la unidad de compensación de movimiento 316 puede generar el bloque de predicción. En este caso, los elementos de sintaxis de información de predicción pueden indicar una imagen de referencia en DPB 314 desde la cual recuperar un bloque de referencia, así como un vector de movimiento que identifica una ubicación del bloque de referencia en la imagen de referencia en relación con la ubicación del bloque actual en la imagen actual. La unidad de compensación de movimiento 316 generalmente puede realizar el proceso de interpredicción de una manera que es sustancialmente similar a la descrita con respecto a la unidad de compensación de movimiento 224 (Figura 3).

Como otro ejemplo, si los elementos de sintaxis de información de predicción indican que el bloque actual está intrapredicho, la unidad de intrapredicción 318 puede generar el bloque de predicción de acuerdo con un modo de intrapredicción indicado por los elementos de sintaxis de

información de predicción. De nuevo, la unidad de intrapredicción 318 puede realizar generalmente el proceso de intrapredicción de una manera que es sustancialmente similar a la descrita con respecto a la unidad de intrapredicción 226 (Figura 3). La unidad de intrapredicción 318 puede recuperar datos de muestras vecinas al bloque actual de DPB 314.

La unidad de reconstrucción 310 puede reconstruir el bloque actual usando el bloque de predicción y el bloque residual. Por ejemplo, la unidad de reconstrucción 310 puede añadir muestras del bloque residual a muestras correspondientes del bloque de predicción para reconstruir el bloque actual.

La unidad de filtro 312 puede realizar una o más operaciones de filtro en bloques reconstruidos. Por ejemplo, la unidad de filtro 312 puede realizar operaciones de desbloqueo para reducir los artefactos de fragmentación en bloques a lo largo de los bordes de los bloques reconstruidos. Las operaciones de la unidad de filtro 312 no se realizan necesariamente en todos los ejemplos.

El decodificador de video 300 puede almacenar los bloques reconstruidos en DPB 314. Por ejemplo, en ejemplos en los que no se realizan operaciones de la unidad de filtro 312, la unidad de reconstrucción 310 puede almacenar bloques reconstruidos en DPB 314. En ejemplos donde se realizan operaciones de la unidad de filtro 312, la unidad de filtro

312 puede almacenar los bloques reconstruidos filtrados en
DPB 314. Como se analizó anteriormente, DPB 314 puede
proporcionar información de referencia, tal como muestras de
una imagen actual para intrapredicción e imágenes previamente
5 decodificadas para compensación de movimiento posterior, a
la unidad de procesamiento de predicción 304. Además, el
decodificador de video 300 puede emitir imágenes
decodificadas (por ejemplo, video decodificado) desde DPB
314 para su posterior presentación en un dispositivo de
10 visualización, tal como el dispositivo de visualización 118
de la Figura 1.

De esta manera, el decodificador de video 300
representa un ejemplo de un dispositivo de decodificación de
video que incluye una memoria configurada para almacenar
15 datos de video, y una o más unidades de procesamiento
implementadas en circuitos y configuradas para generar un
primer ordenamiento de imágenes de referencia en una lista
de imágenes de referencia para un primer bloque y generar un
segundo ordenamiento de las imágenes de referencia en la
20 lista de imágenes de referencia para un segundo bloque, en
donde el primer ordenamiento es diferente del segundo
ordenamiento.

La Figura 5 es un diagrama de flujo que ilustra
un proceso de ejemplo para codificar un bloque actual de
25 acuerdo con las técnicas de esta divulgación. El bloque
actual puede comprender una CU actual. Aunque se describe

con respecto al codificador de video 200 (Figuras 1 y 3), debe entenderse que otros dispositivos pueden configurarse para realizar un proceso similar al de la Figura 5.

En este ejemplo, el codificador de video 200 predice inicialmente el bloque actual (350). Por ejemplo, el codificador de video 200 puede formar un bloque de predicción para el bloque actual. El codificador de video 200 puede entonces calcular un bloque residual para el bloque actual (352). Para calcular el bloque residual, el codificador de video 200 puede calcular una diferencia entre el bloque original, no codificado y el bloque de predicción para el bloque actual. El codificador de video 200 puede entonces transformar el bloque residual y cuantificar los coeficientes de transformada del bloque residual (354). A continuación, el codificador de video 200 puede escanear los coeficientes de transformada cuantificados del bloque residual (356). Durante el escaneo, o después del escaneo, el codificador de video 200 puede codificar por entropía los coeficientes de transformada (358). Por ejemplo, el codificador de video 200 puede codificar los coeficientes de transformada usando CAVLC o CABAC. El codificador de video 200 puede entonces emitir los datos codificados por entropía del bloque (360).

La Figura 6 es un diagrama de flujo que ilustra un proceso de ejemplo para decodificar un bloque actual de datos de video de acuerdo con las técnicas de esta divulgación. El bloque actual puede comprender una CU

actual. Aunque se describe con respecto al decodificador de video 300 (Figuras 1 y 4), debe entenderse que otros dispositivos pueden configurarse para realizar un proceso similar al de la Figura 6.

5 El decodificador de video 300 puede recibir datos codificados por entropía para el bloque actual, tales como información de predicción codificada por entropía y datos codificados por entropía para coeficientes de transformada de un bloque residual correspondiente al bloque actual
10 (370). El decodificador de video 300 puede decodificar por entropía los datos codificados por entropía para determinar la información de predicción para el bloque actual y para reproducir coeficientes de transformada del bloque residual (372). El decodificador de video 300 puede predecir el
15 bloque actual (374), por ejemplo, usando un modo de intra o interpredicción como se indica por la información de predicción para el bloque actual, para calcular un bloque de predicción para el bloque actual. El decodificador de video 300 puede entonces escanear de manera inversa los
20 coeficientes de transformada reproducidos (376), para crear un bloque de coeficientes de transformada cuantificados. El decodificador de video 300 puede entonces cuantificar de manera inversa los coeficientes de transformada y aplicar una transformada inversa a los coeficientes de
25 transformada para producir un bloque residual (378). El decodificador de video 300 puede decodificar en última

instancia el bloque actual al combinar el bloque de predicción y el bloque residual (380).

La Figura 7 es un diagrama de flujo que ilustra un proceso de ejemplo para decodificar bloques de datos de video de acuerdo con las técnicas de esta divulgación. Los bloques pueden comprender una CU. Aunque se describe con respecto al decodificador de video 300 (Figuras 1 y 4), debe entenderse que otros dispositivos pueden configurarse para realizar un proceso similar al de la Figura 7. Además, debe entenderse que un bucle de decodificación de video de un codificador de video, tal como el codificador de video 200, también puede realizar el proceso de la Figura 7.

En el ejemplo de la Figura 7, el decodificador de video 300 genera un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte (400). Para generar el primer ordenamiento de las imágenes de referencia para el primer bloque, el decodificador de video 300 puede asignar índices a las imágenes de referencia.

El decodificador de video 300 decodifica el primer bloque usando el primer ordenamiento de las imágenes de referencia (402). Para decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia, el decodificador de video 300 puede, por ejemplo, determinar una primera imagen de referencia para el primer bloque basado en el primer ordenamiento de las imágenes de referencia e

interpretar el primer bloque a partir de un bloque en la primera imagen de referencia. Para interpretar el primer bloque, el decodificador de video 300 puede, por ejemplo, determinar un vector de movimiento usando AMVP y después usar el vector de movimiento para localizar un bloque de predicción en la primera imagen de referencia.

El decodificador de video 300 genera un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de un proceso de reordenamiento de imágenes de referencia adaptable (404). El primer bloque y el segundo bloque pueden, por ejemplo, pertenecer a una misma imagen o corte y pueden ser bloques espacialmente vecinos. El primer pedido es diferente del segundo pedido. Para generar el segundo ordenamiento de imágenes de referencia para el segundo bloque, el decodificador de video 300 asigna al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento. En algunos ejemplos, uno o ambos del primer y segundo bloques pueden ser bloques bipredichos, en cuyo caso el decodificador de video 300 mantiene dos listas para el bloque y determina un bloque de predicción final basado en dos bloques de predicción a partir de dos imágenes de referencia diferentes.

Como se describió anteriormente, en algunos ejemplos, el proceso de reordenamiento de imágenes de referencia adaptable puede incluir un proceso de

emparejamiento de plantilla. En algunos ejemplos, para bloques bipredichos, el proceso de reordenamiento de imágenes de referencia adaptable puede incluir un proceso de emparejamiento bilateral. En algunos ejemplos, el
5 proceso de reordenamiento de imágenes de referencia adaptable puede incluir comparar un recuento de orden de imágenes (POC) de una imagen en la primera lista de imágenes de referencia con un POC de una imagen en la segunda lista de imágenes de referencia.

10 El decodificador de video 300 decodifica el segundo bloque usando el segundo ordenamiento de las imágenes de referencia (406). Al igual que con el primer bloque, para decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia, el decodificador
15 de video 300 puede, por ejemplo, determinar una segunda imagen de referencia para el segundo bloque en función del segundo ordenamiento de las imágenes de referencia e interpredecir el segundo bloque a partir de un bloque en la segunda imagen de referencia. Para interpredecir el segundo
20 bloque, el decodificador de video 300 puede, por ejemplo, determinar un vector de movimiento usando AMVP y después usar el vector de movimiento para localizar un bloque de predicción en la segunda imagen de referencia.

Para decodificar el segundo bloque usando el
25 segundo ordenamiento de las imágenes de referencia, el decodificador de video 300 puede, por ejemplo, recibir un

primer elemento de sintaxis que identifica un predictor de vector de movimiento, un segundo elemento de sintaxis que identifica un MVD; y un tercer elemento de sintaxis que identifica un índice de una imagen de referencia en la lista de imágenes de referencia. El decodificador de video 300 determina un vector de movimiento al añadir el predictor del vector de movimiento al MVD. El decodificador de video 300 puede identificar una imagen de referencia en la lista de imágenes de referencia en función del índice y el segundo ordenamiento de las imágenes de referencia, al localizar un bloque de predicción en la imagen de referencia basado en el vector de movimiento, y decodificar el segundo bloque usando el bloque de predicción.

El decodificador de video 300 emite datos de video decodificados que comprenden versiones descodificadas del primer bloque y del segundo bloque (408). El decodificador de video 300 puede, por ejemplo, emitir imágenes de datos de video decodificados para su almacenamiento, transmisión o visualización. En ejemplos, donde el proceso de la Figura 7 es realizado por un codificador de video, el codificador de video puede emitir los datos de video decodificados para su almacenamiento en un búfer de imágenes decodificadas de manera que las imágenes decodificadas puedan usarse para codificar otras imágenes de los datos de video.

Las siguientes cláusulas numeradas ilustran uno o más aspectos de los dispositivos y técnicas descritos en esta divulgación.

Cláusula 1A. Un método de codificación de datos de video, en donde el método comprende: generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque; y generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque, en donde el primer ordenamiento es diferente del segundo ordenamiento.

Cláusula 2A. El método de la cláusula 1A, en donde el primer bloque y el segundo bloque pertenecen a un mismo corte.

Cláusula 3A. El método de la cláusula 1A o 2A, en donde el primer bloque y el segundo bloque pertenecen a una misma imagen y son bloques espacialmente vecinos.

Cláusula 4A. El método de cualquiera de las cláusulas 1A-3A, en donde: generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento.

Cláusula 5A. El método de cualquiera de las cláusulas 1A-3A, en donde: generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; y
5 generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento en función de si se ha escalado o no una imagen de referencia.

10 Cláusula 6A. El método de la cláusula 1A, en donde: generar el primer ordenamiento de imágenes de referencia en la lista de imágenes de referencia para el primer bloque comprende generar el primer ordenamiento en función de la aplicación de un proceso de emparejamiento
15 de plantilla a las imágenes de referencia en la lista de imágenes de referencia; y generar el segundo ordenamiento de imágenes de referencia en la lista de imágenes de referencia para el segundo bloque comprende generar el segundo ordenamiento en función de la aplicación del
20 proceso de emparejamiento de plantilla a las imágenes de referencia en la lista de imágenes de referencia.

Cláusula 7A. El método de cualquiera de las cláusulas 1A-6A, en donde el segundo bloque comprende un bloque bipredicho.

25 Cláusula 8A. El método de la cláusula 7A, en donde la lista de imágenes de referencia es una primera lista de

imágenes de referencia, en donde el método comprende además mantener una segunda lista de imágenes de referencia.

Cláusula 9A. El método de la cláusula 8A, en donde generar el segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el
5 segundo bloque comprende comparar un recuento de orden de imágenes (POC) de una imagen en la primera lista de imágenes de referencia con un POC de una imagen en la segunda lista de imágenes de referencia.

10 Cláusula 10A. El método de cualquiera de las cláusulas 1A-9A, que comprende además: determinar una primera imagen de referencia para el primer bloque en función del primer ordenamiento de las imágenes de referencia; interpredecir el primer bloque en función de
15 la primera imagen de referencia; determinar una segunda imagen de referencia para el segundo bloque en función del segundo ordenamiento de las imágenes de referencia; interpredecir el segundo bloque en función de la segunda imagen de referencia.

20 Cláusula 11A. El método de 10A, que comprende además: determinar una predicción de signos para una diferencia de vector de movimiento en función de la generación del segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el
25 segundo bloque.

Cláusula 12A. El método de cualquiera de las cláusulas 1A-10A, que comprende además: deshabilitar la predicción de signos en función de una condición.

5 Cláusula 13A. El método de cualquiera de las cláusulas 1A-10A, que comprende además: en respuesta a una serie de signos predichos que exceden un número umbral, signos subsiguientes de codificación regular.

Cláusula 14A. Un método de codificación de datos de video, en donde el método comprende: para una pluralidad
10 de bloques de datos de video, determinar que el remuestreo de imágenes de referencia está habilitado; en respuesta a la determinación de que ese remuestreo de imágenes de referencia está habilitado, determinar que el reordenamiento de imágenes de referencia está deshabilitado.

15 Cláusula 15A. El método de la cláusula 14A, en donde determinar que el remuestreo de imágenes de referencia está habilitado comprende recibir un elemento de sintaxis y determinar que el reordenamiento de imágenes de referencia está deshabilitado comprende inferir, en
20 función de un valor del elemento de sintaxis, que el reordenamiento de imágenes de referencia está deshabilitado.

Cláusula 16A. El método de cualquiera de las cláusulas 1A-15A, en donde la codificación comprende
25 codificación.

Cláusula 17A. El método de cualquiera de las cláusulas 1A-15A, en donde la codificación comprende decodificación.

Cláusula 18A. Un dispositivo para codificar datos
5 de video, cuyo dispositivo comprende uno o más medios para realizar el método de cualquiera de las cláusulas 1A-17A.

Cláusula 19A. El dispositivo de la cláusula 18A, en donde uno o más medios comprenden uno o más procesadores implementados en circuitos.

10 Cláusula 20A. El dispositivo de cualquiera de las cláusulas 18A y 19A, que comprende además una memoria para almacenar los datos de video.

Cláusula 21A. El dispositivo de cualquiera de las cláusulas 18A-20A, que comprende además una pantalla
15 configurada para mostrar datos de video decodificados.

Cláusula 22A. El dispositivo de cualquiera de las cláusulas 18A-21A, en donde el dispositivo comprende uno o más de una cámara, un computador, un dispositivo móvil, un dispositivo receptor de difusión o un decodificador.

20 Cláusula 23A. El dispositivo de cualquiera de las cláusulas 18A-22A, en donde el dispositivo comprende un decodificador de video.

Cláusula 24A. El dispositivo de cualquiera de las cláusulas 18A-23A, en donde el dispositivo comprende
25 un codificador de video.

Cláusula 25A. Un medio de almacenamiento legible por computador que tiene almacenadas en este instrucciones que, cuando son ejecutadas, hacen que uno o más procesadores realicen el método de cualquiera de las cláusulas 1A-17A.

5 Cláusula 1B: Un método de decodificación de datos de video, en donde el método comprende: generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices
10 a las imágenes de referencia; decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de
15 un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende
20 asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento; decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y emitir datos de video decodificados que comprenden versiones
25 decodificadas del primer bloque y del segundo bloque.

Cláusula 2B: El método de la cláusula 1B, en donde el proceso de reordenamiento de imágenes de referencia adaptable comprende un proceso de emparejamiento de plantilla, en donde el método comprende
5 además: determinar un costo de emparejamiento de plantilla para al menos una primera imagen de referencia y una segunda imagen de referencia en la lista de imágenes de referencia; y generar el segundo ordenamiento de las imágenes de referencia en la lista de imágenes de
10 referencia para el segundo bloque del corte en función de una comparación de un costo de emparejamiento de plantilla para la primera imagen de referencia con un costo de emparejamiento de plantilla para la segunda imagen de referencia.

15 Cláusula 3B: El método de la cláusula 1B o 2B, en donde el proceso de reordenamiento de imágenes de referencia adaptable comprende un proceso de emparejamiento bilateral.

Cláusula 4B: El método de cualquiera de las cláusulas 1B-3B, en donde el proceso de reordenamiento de
20 imágenes de referencia adaptable comprende comparar un recuento de orden de imágenes (POC) de una primera imagen en la lista de imágenes de referencia con un POC de una segunda imagen en la lista de imágenes de referencia.

Cláusula 5B: El método de cualquiera de las
25 cláusulas 1B-4B, en donde el primer bloque y el segundo

bloque pertenecen a una misma imagen y son bloques espacialmente vecinos.

Cláusula 6B: El método de cualquiera de las cláusulas 1B-5B, en donde el segundo bloque comprende un
5 bloque bipredicho y la lista de imágenes de referencia es una primera lista de imágenes de referencia, en donde el método comprende además: mantener una segunda lista de imágenes de referencia; y decodificar el segundo bloque usando la primera lista de imágenes de referencia y la
10 segunda lista de imágenes de referencia.

Cláusula 7B: El método de cualquiera de las cláusulas 1B-6B, en donde la decodificación del primer bloque usando el primer ordenamiento de las imágenes de referencia comprende: determinar una primera imagen de
15 referencia para el primer bloque en función del primer ordenamiento de las imágenes de referencia; interpredecir el primer bloque en función de la primera imagen de referencia; en donde la decodificación del segundo bloque usando el segundo ordenamiento de las imágenes de
20 referencia comprende: determinar una segunda imagen de referencia para el segundo bloque en función del segundo ordenamiento de las imágenes de referencia; interpredecir el segundo bloque en función de la segunda imagen de referencia.

25 Cláusula 8B: El método de cualquiera de las cláusulas 1B-7B, que comprende además: determinar una

predicción de signos para una diferencia de vector de movimiento en función de la generación del segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el segundo bloque.

5 Cláusula 9B: El método de cualquiera de las cláusulas 1B-8B, que comprende además: recibir un primer elemento de sintaxis que identifica un predictor de vector de movimiento; recibir un segundo elemento de sintaxis que identifica una diferencia de vector de movimiento; y
10 recibir un tercer elemento de sintaxis que identifica un índice de una imagen de referencia en la lista de imágenes de referencia; identificar una imagen de referencia en la lista de imágenes de referencia en función del índice y el segundo ordenamiento de las imágenes de referencia;
15 localizar un bloque de predicción en la imagen de referencia en función del predictor del vector de movimiento y la diferencia del vector de movimiento; decodificar el segundo bloque usando el bloque de predicción.

20 Cláusula 10B: El método de cualquiera de las cláusulas 1B-9B, que comprende además: determinar que el segundo bloque se codifica usando un modo predictor de vector de movimiento avanzado.

 Cláusula 11B: El método de cualquiera de las
25 cláusulas 1B-8B o 10B, en donde el método de decodificación se realiza como parte de un proceso de codificación de video.

Cláusula 12B: Un dispositivo para decodificar datos de video, en donde el dispositivo comprende: una memoria configurada para almacenar datos de video; uno o más procesadores implementados en circuitos y configurados para: generar un primer ordenamiento de imágenes de referencia en una lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de un proceso de reordenamiento de imágenes de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprende asignar al menos algunos de los índices a imágenes de referencia diferentes que en el primer ordenamiento; decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del segundo bloque.

Cláusula 13B: El dispositivo de la cláusula 12B, en donde el proceso de reordenamiento de imágenes de

referencia adaptable comprende un proceso de emparejamiento de plantilla, y el uno o más procesadores están configurados además para: determinar un costo de emparejamiento de plantilla para al menos una primera imagen de referencia y
5 una segunda imagen de referencia en la lista de imágenes de referencia; y generar el segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el segundo bloque del corte en función de una comparación de un costo de emparejamiento de plantilla para
10 la primera imagen de referencia con un costo de emparejamiento de plantilla para la segunda imagen de referencia.

Cláusula 14B: El dispositivo de la cláusula 12B o 13B, en donde el proceso de reordenamiento de imágenes
15 de referencia adaptable comprende un proceso de emparejamiento bilateral.

Cláusula 15B: El dispositivo de cualquiera de las cláusulas 12B-14B, en donde el proceso de reordenamiento de imágenes de referencia adaptable comprende comparar un
20 recuento de orden de imágenes (POC) de una primera imagen en la lista de imágenes de referencia con un POC de una segunda imagen en la lista de imágenes de referencia.

Cláusula 16B: El dispositivo de cualquiera de las cláusulas 12B-15B, en donde el primer bloque y el segundo
25 bloque pertenecen a una misma imagen y son bloques espacialmente vecinos.

Cláusula 17B: El dispositivo de cualquiera de las cláusulas 12B-16B, en donde el segundo bloque comprende un bloque bipredicho y la lista de imágenes de referencia es una primera lista de imágenes de referencia, y el uno o más procesadores están configurados además para: mantener una segunda lista de imágenes de referencia; y decodificar el segundo bloque usando la primera lista de imágenes de referencia y la segunda lista de imágenes de referencia.

Cláusula 18B: El dispositivo de cualquiera de las cláusulas 12B-17B, en donde, para decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia, el uno o más procesadores están configurados además para: determinar una primera imagen de referencia para el primer bloque en función del primer ordenamiento de las imágenes de referencia; interpredecir el primer bloque en función de la primera imagen de referencia; en donde, para decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia, el uno o más procesadores están configurados además para: determinar una segunda imagen de referencia para el segundo bloque en función del segundo ordenamiento de las imágenes de referencia; interpredecir el segundo bloque en función de la segunda imagen de referencia.

Cláusula 19B: El dispositivo de cualquiera de las cláusulas 12B-18B, en donde el uno o más procesadores están configurados además para: determinar una predicción

de signos para una diferencia de vector de movimiento en función de la generación del segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el segundo bloque.

5 Cláusula 20B: El dispositivo de cualquiera de las cláusulas 12B-19B, en donde el uno o más procesadores están configurados además para: recibir un primer elemento de sintaxis que identifica un predictor de vector de movimiento; recibir un segundo elemento de sintaxis que
10 identifica una diferencia de vector de movimiento; y recibir un tercer elemento de sintaxis que identifica un índice de una imagen de referencia en la lista de imágenes de referencia; identificar una imagen de referencia en la lista de imágenes de referencia en función del índice y el
15 segundo ordenamiento de las imágenes de referencia; localizar un bloque de predicción en la imagen de referencia en función del predictor del vector de movimiento y la diferencia del vector de movimiento; decodificar el segundo bloque usando el bloque de
20 predicción.

 Cláusula 21B: El dispositivo de cualquiera de las cláusulas 1B-202B, en donde el uno o más procesadores están configurados además para: determinar que el segundo bloque se codifica usando un modo de predictor de vector de
25 movimiento avanzado.

Cláusula 22B: El dispositivo de cualquiera de las cláusulas 12B-21B, en donde el dispositivo comprende un dispositivo de comunicación inalámbrica, que comprende además un receptor configurado para recibir datos de video
5 codificados.

Cláusula 23B: El dispositivo de la cláusula 22B, en donde el dispositivo de comunicación inalámbrica comprende un terminal telefónico y en donde el receptor está configurado para demodular, de acuerdo con un estándar
10 de comunicación inalámbrica, una señal que comprende los datos de video codificados.

Cláusula 24B: El dispositivo de cualquiera de las cláusulas 12B-23B, que comprende además: una pantalla configurada para visualizar datos de video decodificados.

15 Cláusula 25B: El dispositivo de cualquiera de las cláusulas 12B-24B, en donde el dispositivo comprende uno o más de una cámara, un computador, un dispositivo móvil, un dispositivo receptor de difusión o un decodificador.

20 Cláusula 26B: El dispositivo de cualquiera de las cláusulas 12B a 25B, en donde el dispositivo comprende una cámara.

Cláusula 27B: Un medio de almacenamiento legible por computador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores, hacen que el uno o
25 más procesadores: generen un primer ordenamiento de imágenes de referencia en una lista de imágenes de

referencia para un primer bloque de un corte, en donde
generar el primer ordenamiento de las imágenes de
referencia para el primer bloque comprende asignar índices
a las imágenes de referencia; generen un segundo
5 ordenamiento de las imágenes de referencia en la lista de
imágenes de referencia para un segundo bloque del corte en
función de un proceso de reordenamiento de imágenes de
referencia adaptable, en donde el primer ordenamiento es
diferente del segundo ordenamiento y generar el segundo
10 ordenamiento de imágenes de referencia para el segundo
bloque comprende asignar al menos algunos de los índices
a imágenes de referencia diferentes que en el primer
ordenamiento; decodifiquen el primer bloque usando el
primer ordenamiento de las imágenes de referencia;
15 decodifiquen el segundo bloque usando el segundo
ordenamiento de las imágenes de referencia; y emitan datos
de video decodificados que comprenden versiones
decodificadas del primer bloque y del segundo bloque.

Cláusula 28B: El medio de almacenamiento legible
20 por computador de la cláusula 27B, en donde el proceso de
reordenamiento de imágenes de referencia adaptable
comprende un proceso de emparejamiento de plantilla, y las
instrucciones hacen además que el uno o más procesadores:
determinen un costo de emparejamiento de plantilla para al
25 menos una primera imagen de referencia y una segunda imagen
de referencia en la lista de imágenes de referencia; y

generen el segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para el segundo bloque del corte en función de una comparación de un costo de emparejamiento de plantilla para la primera
5 imagen de referencia con un costo de emparejamiento de plantilla para la segunda imagen de referencia.

Cláusula 29B: El medio de almacenamiento legible por computador de la cláusula 27B o 28B, en donde el proceso de reordenamiento de imágenes de referencia adaptable
10 comprende un proceso de emparejamiento bilateral.

Cláusula 30B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-29B, en donde el proceso de reordenamiento de imágenes de referencia adaptable comprende comparar un recuento de
15 orden de imágenes (POC) de una primera imagen en la lista de imágenes de referencia con un POC de una segunda imagen en la lista de imágenes de referencia.

Cláusula 31B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-30B, en
20 donde el primer bloque y el segundo bloque pertenecen a una misma imagen y son bloques espacialmente vecinos.

Cláusula 32B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-31B, en donde el segundo bloque comprende un bloque bipredicho y
25 la lista de imágenes de referencia es una primera lista de imágenes de referencia, y las instrucciones hacen además

que el uno o más procesadores: mantengan una segunda lista de imágenes de referencia; y decodifiquen el segundo bloque usando la primera lista de imágenes de referencia y la segunda lista de imágenes de referencia.

5 Cláusula 33B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-32B, en donde, para decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia, las instrucciones hacen además que el uno o más procesadores estén configurados
10 además para: determinar una primera imagen de referencia para el primer bloque en función del primer ordenamiento de las imágenes de referencia; interpredecir el primer bloque en función de la primera imagen de referencia; en donde, para decodificar el segundo bloque usando el segundo ordenamiento
15 de las imágenes de referencia, las instrucciones hacen además que el uno o más procesadores: determinen una segunda imagen de referencia para el segundo bloque en función del segundo ordenamiento de las imágenes de referencia; interpretengan el segundo bloque en función de la segunda imagen de referencia.

20 Cláusula 34B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-33B, en donde las instrucciones hacen además que el uno o más procesadores: determinen una predicción de signos para una diferencia de vector de movimiento en función de la
25 generación del segundo ordenamiento de las imágenes de

referencia en la lista de imágenes de referencia para el segundo bloque.

Cláusula 35B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-34B, en
5 donde las instrucciones hacen además que el uno o más procesadores: reciban un primer elemento de sintaxis que identifica un predictor de vector de movimiento; reciban un segundo elemento de sintaxis que identifica una diferencia de vector de movimiento; y reciban un tercer elemento de
10 sintaxis que identifica un índice de una imagen de referencia en la lista de imágenes de referencia; identifiquen una imagen de referencia en la lista de imágenes de referencia en función del índice y el segundo ordenamiento de las imágenes de referencia; localicen un bloque de predicción en
15 la imagen de referencia en función del predictor del vector de movimiento y la diferencia del vector de movimiento; decodifiquen el segundo bloque usando el bloque de predicción.

Cláusula 36B: El medio de almacenamiento legible por computador de cualquiera de las cláusulas 27B-35B, en
20 donde las instrucciones hacen además que el uno o más procesadores: determinen que el segundo bloque se codifica usando un modo de predictor de vector de movimiento avanzado.

Cláusula 37B: Un aparato para decodificar datos
25 de video, en donde el aparato comprende: medios para generar un primer ordenamiento de imágenes de referencia en una

lista de imágenes de referencia para un primer bloque de un corte, en donde generar el primer ordenamiento de las imágenes de referencia para el primer bloque comprende asignar índices a las imágenes de referencia; medios para
5 decodificar el primer bloque usando el primer ordenamiento de las imágenes de referencia; medios para generar un segundo ordenamiento de las imágenes de referencia en la lista de imágenes de referencia para un segundo bloque del corte en función de un proceso de reordenamiento de imágenes
10 de referencia adaptable, en donde el primer ordenamiento es diferente del segundo ordenamiento y los medios para generar el segundo ordenamiento de imágenes de referencia para el segundo bloque comprenden medios para asignar al menos algunos de los índices a imágenes de referencia
15 diferentes que en el primer ordenamiento; medios para decodificar el segundo bloque usando el segundo ordenamiento de las imágenes de referencia; y medios para emitir datos de video decodificados que comprenden versiones decodificadas del primer bloque y del
20 segundo bloque.

Se debe reconocer que, según el ejemplo, ciertas acciones o eventos de cualquiera de las técnicas descritas en la presente pueden realizarse en una secuencia diferente, pueden añadirse, fusionarse o dejarse fuera por
25 completo (por ejemplo, no todas las acciones o eventos descritos son necesarios para poner en práctica las

técnicas). Además, en ciertos ejemplos, las acciones o eventos pueden realizarse simultáneamente, por ejemplo, a través de procesamiento multiproceso, procesamiento de interrupción o múltiples procesadores, en lugar de
5 secuencialmente.

En uno o más ejemplos, las funciones descritas pueden implementarse en *hardware*, *software*, *firmware* o cualquier combinación de estos. Si se implementan en *software*, las funciones pueden almacenarse o transmitirse
10 como una o más instrucciones o código en un medio legible por computador y ser ejecutadas por una unidad de procesamiento basada en *hardware*. Los medios legibles por computador pueden incluir medios de almacenamiento legibles por computador, que corresponden a un medio tangible tal
15 como medios de almacenamiento de datos, o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por computador
20 generalmente pueden corresponder a (1) medios de almacenamiento legibles por computador tangibles que no son transitorios o (2) un medio de comunicación tal como una señal u onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que pueda
25 acceder uno o más computadores 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 computador.

A manera de ejemplo, y no de limitación, dichos
5 medios de almacenamiento legibles por computador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, memoria flash, o cualquier otro medio que pueda usarse para almacenar el
10 código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda acceder un computador. Además, cualquier conexión se denomina adecuadamente medio legible por computador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, servidor
15 u otra fuente remota usando un cable coaxial, cable de fibra óptica, par trenzado, línea de suscriptor digital (DSL) o tecnologías inalámbricas tales como infrarroja, radio y microondas, el cable coaxial, cable de fibra óptica, par trenzado, DSL, o tecnologías inalámbricas tales como
20 infrarroja, radio y microondas se incluyen en la definición de medio. Sin embargo, debe entenderse que los medios de almacenamiento legibles por computador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que
25 se refieren a medios de almacenamiento tangibles no transitorios. Disco, como se usa en la presente, incluye

disco compacto (CD, por sus siglas en inglés), disco láser, disco óptico, disco versátil digital (DVD, por sus siglas en inglés), disquete y disco Blu-ray, donde los discos generalmente reproducen datos magnéticamente u ópticamente con láseres. Las combinaciones de los anteriores también deben incluirse dentro del alcance de los medios legibles por computador.

Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más DSP, microprocesadores de propósito general, ASIC, FPGA u otros circuitos lógicos integrados o discretos equivalentes. Por consiguiente, los términos “procesador” y “circuito de procesamiento”, tal como se usan en la presente, pueden referirse a cualquiera de las estructuras anteriores o cualquier otra estructura adecuada para la implementación de las técnicas descritas en la presente. Además, en algunos aspectos, la funcionalidad descrita en la presente puede proporcionarse dentro de módulos de *hardware* y/o *software* dedicados configurados para codificar y decodificar, o incorporarse en un códec combinado. Asimismo, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

Las técnicas de esta divulgación pueden implementarse en una amplia variedad de dispositivos o aparatos, incluido un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de CI (por ejemplo, un conjunto de chips). En esta divulgación se describen

diversos componentes, módulos o unidades para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no necesariamente requieren realización por diferentes unidades de *hardware*.

5 Más bien, como se describió anteriormente, varias unidades pueden combinarse en una unidad de *hardware* de códec o ser proporcionadas por una colección de unidades de *hardware* interoperativas, incluido uno o más procesadores descritos anteriormente, junto con *software* y/o *firmware* adecuados.

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