

MODO DE PREDICCIÓN INTER-INTRA PARA DATOS DE VÍDEO

[0001] Esta solicitud reivindica prioridad a la solicitud estadounidense N° **16/781.751**, presentada el 4 de febrero de 2020, y la solicitud provisional estadounidense N° **62/802.515**, presentada el 7 de febrero 2019, cuyos contenidos completos se incorporan en la presente por referencia.

CAMPO TÉCNICO

[0002] Esta divulgación se refiere a la codificación de video, incluida la codificación y la decodificación de video.

ANTECEDENTES

[0003] Se pueden incorporar capacidades de video digitales en una amplia rango de dispositivos, lo que incluye televisores digitales, sistemas de transmisión directa digital, sistemas de transmisión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores tipo tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de registro digital, reproductores de medios digitales, dispositivos de juegos de video, consolas de juegos de video, teléfonos celulares o de radio satélite, los denominados "teléfonos inteligentes", dispositivos de teleconferencia de video, dispositivos de streaming de video, y similares. Los dispositivos de video digital implementan técnicas de codificación de video, tales como aquellas 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), ITU-T H.265/codificación de video de alta eficiencia (HEVC) y las extensiones de estos estándares. Los dispositivos de video pueden transmitir, recibir, codificar, decodificar y/o almacenar información de video digital de manera más eficaz implementando las técnicas de codificación de video.

[0004] Las técnicas de codificación de video incluyen la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) para reducir o eliminar la redundancia inherente en secuencias de video. Para la codificación de video basado en bloques, un plano de video (por ejemplo, una imagen de video o una porción de una imagen de video) se pueden dividir en bloques de video, que también pueden ser referidos como unidades de árboles de codificación (CTU), unidades de codificación (CU) y/o nodos de codificación. Los bloques de video en un (I) porción intracodificada de una imagen se codifican mediante el uso de predicción espacial con respecto a las muestras de referencia en bloques vecinos de la misma imagen. Los bloques de video

en una porción intercodificada (P o B) de la imagen pueden usar predicción espacial con respecto a las muestras de referencia en bloques vecinos de la misma imagen o predicción temporal con respecto a las muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse cuadros, imágenes de referencia y pueden denominarse cuadros de referencia.

SÍNTESIS

[0005] En general, esta divulgación describe técnicas para realizar predicciones inter-intra combinadas. Es decir, en la codificación de video, un codificador de video (por ejemplo, un codificador de video o un decodificador de video) puede realizar una codificación a base de bloques de imágenes que forman una secuencia de video. El codificador de video puede codificar bloques de la imagen mediante la formación de un bloque de predicción y la codificación de un bloque residual que representa las diferencias muestra por muestra entre el bloque de predicción y el bloque real a codificar. La interpredicción representa la predicción de un bloque de referencia de una imagen previamente codificada, mientras que la intrapredicción representa la predicción de datos de referencia previamente codificados de la imagen actual. La predicción inter-intra combinada representa la predicción que utiliza tanto la interpredicción como la intrapredicción. En particular, esta divulgación describe técnicas mejoradas para realizar predicciones inter-intra combinadas. Las técnicas de esta divulgación se pueden usar, por ejemplo, en codificación de video versátil (VVC) u otros estándares de codificación de video que soportan la predicción inter-intra combinada.

[0006] En un ejemplo, un método de codificación de datos de video incluye formar un bloque de interpredicción para un bloque de crominancia actual de datos de video; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el bloque de predicción.

[0007] En otro ejemplo, un dispositivo para codificar datos de video incluye una

memoria para almacenar datos de video y uno o más procesadores implementados en circuitos y configurados para formar un bloque de interpredicción para un bloque de crominancia actual de los datos de video; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el bloque de predicción.

[0008] En otro ejemplo, un medio de almacenamiento legible por ordenador tiene almacenadas en este instrucciones que, cuando se ejecutan, hacen que un procesador forme un bloque de interpredicción para un bloque de crominancia actual de datos de vídeo; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el bloque de predicción.

[0009] En otro ejemplo, un dispositivo de codificación de datos de video incluye medios para formar un bloque de interpredicción para un bloque de crominancia actual de datos de video; medios para formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; medios para determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; medios para determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; medios para aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; medios para combinar el primer bloque de interpredicción ponderado y

el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y medios para codificar el bloque de crominancia actual mediante el bloque de predicción.

[0010] Los detalles de uno o más ejemplos se exponen en las figuras adjuntas y en la descripción siguiente. Otras características, objetivos y ventajas serán evidentes a partir de la descripción, figuras y reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0011] La Figura 1 es un diagrama de bloques que ilustra un ejemplo de sistema de codificación y decodificación de video que puede realizar técnicas de la presente divulgación.

[0012] Las Figuras 2A y 2B son diagramas conceptuales que ilustran un ejemplo de estructura de árbol binario de árbol cuaternario (QTBT) y una unidad de árbol de codificación (CTU) correspondiente.

[0013] Las Figuras 3A-3F son diagramas conceptuales que ilustran ubicaciones de ejemplo de bloques vecinos a bloques actuales.

[0014] La Figura 4 es un diagrama de bloques que ilustra un codificador de video de ejemplo que puede realizar técnicas de la presente divulgación.

[0015] La Figura 5 es un diagrama de bloques que ilustra un decodificador de video que puede realizar técnicas de la presente divulgación.

[0016] La Figura 6 es un diagrama de flujo que ilustra un procedimiento de ejemplo para la codificación actual de un bloque de acuerdo con las técnicas de esta divulgación.

[0017] La Figura 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo para la decodificación actual de un bloque de acuerdo con las técnicas de esta divulgación.

[0018] La Figura 8 es un diagrama de flujo que ilustra un método de ejemplo de codificación (codificación o decodificación) de datos de video de acuerdo con las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

[0019] Los estándares de codificación de video incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también denominado ISO/IEC MPEG-4 AVC), que incluye sus extensiones de Codificación de Video Escalable (SVC) y Codificación de Video Multivista (MVC).

[0020] La Codificación de Video de Alta Eficiencia (HEVC), fue finalizada por el Equipo

de Colaboración Conjunta sobre Codificación de Video (JCT-VC) del Grupo de Expertos de Codificación de Video ITU-T (VCEG) y Grupo de Expertos de Imágenes en Movimiento ISO/IEC (MPEG) en abril de 2013.

[0021] El *Joint Video Experts Team* (JVET), un equipo colaborativo formado por MPEG e ITU-T Study Group 16's VCEG empezó a trabajar recientemente en un nuevo estándar de codificación de video que se conocerá como Codificación de video versátil (VVC). El objetivo principal de VVC es proporcionar una mejora significativa en el rendimiento de compresión sobre el estándar HEVC existente, ayudar en la implementación de servicios de video de mayor calidad y aplicaciones emergentes como multimedia inmersiva omnidireccional de 360° y video de alto rango dinámico (HDR).

[0022] La divulgación describe técnicas que pueden mejorar el modo de predicción inter-intra para la predicción de datos de video. Se ha demostrado que el modo intra de hipótesis múltiples (MHI) mejora el modo de fusión al combinar una intrapredicción y una predicción indexada de fusión, y se adoptó en la 12ª reunión de JVET Macao: M.-S. Chiang, C.-W. Hsu, Y.-W. Huang, S.-M. Lei "CE10.1.1: Predicción de hipótesis múltiples para mejorar el modo AMVP, el modo de salto o fusión y el modo intra", en JVET-L0100, octubre de 2018. De acuerdo con MHI, los candidatos en un conjunto de 4 modos intra, que incluyen predicciones planas, DC, horizontales y verticales, se evalúan para seleccionar el óptimo para combinar con una predicción indexada de fusión. En el proceso de combinación establecido en el ejemplo anterior, si se selecciona el modo plano o DC, o el tamaño del bloque es menor que 4, se despliegan ponderaciones iguales. De lo contrario, las ponderaciones de las muestras intra e inter predichas (w_{Intra} , w_{Inter}) son diferentes según la región de las muestras dentro del bloque de codificación.

[0023] Esta divulgación reconoce que en las técnicas de predicción inter-intra descritas en JVET-L0100, la información de codificación de bloques vecinos no se ha utilizado para mejorar el rendimiento de codificación de estas herramientas. Además, la ponderación en función de la región se emplea para los modos de predicción horizontal y vertical, lo que puede causar artefactos a lo largo de los límites de la región.

[0024] Las técnicas de esta divulgación pueden mejorar el rendimiento de la codificación inter-intra. Estas mejoras pueden resultar de un esquema de ponderancia independiente de la posición de estas técnicas. De acuerdo con las técnicas de esta divulgación, las ponderaciones para las muestras de predicción inter e intra pueden

determinarse de forma adaptativa mediante el uso de información de codificación, por ejemplo, el número de bloques vecinos intracodificados, el número de interpredicciones de los bloques indexados de fusión (unipredicción o bipredicción) y/o el tamaño del bloque actual.

[0025] La Figura 1 es un diagrama de bloques que ilustra un ejemplo de sistema de codificación 100 y decodificación de video que puede realizar técnicas de la presente divulgación. Las técnicas de esta divulgación están generalmente dirigidas 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 formato, sin codificar, video codificado, video decodificado (por ejemplo, reconstruido) y metadatos de video, como datos de señalización.

[0026] Como se muestra en la Figura 1, el sistema 100 incluye un dispositivo fuente 102 que proporciona datos de video codificados que se desean decodificar y mostrar posteriormente mediante un dispositivo de destino 116, en este ejemplo. En particular, el dispositivo fuente 102 proporciona los datos de video al dispositivo de destino 116 mediante un medio legible por computadora 110. El dispositivo fuente 102 o dispositivo de destino 116 puede comprender cualquiera de un amplio rango de dispositivos, lo que incluye ordenadores de escritorio, ordenadores portátiles (es decir, laptop), ordenadores tipo tableta, decodificadores, aparatos telefónicos, tales como teléfonos inteligentes, televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de juegos de video, dispositivo de streaming de video, o similares. En algunos casos, el dispositivo fuente 102 y el dispositivo de destino 116 pueden estar equipados para comunicación inalámbrica, y por lo tanto pueden denominarse dispositivos de comunicación inalámbrica.

[0027] 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 la presente 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 de codificación inter-intra. 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 de

una fuente de video externa, tal como una cámara externa. Asimismo, el dispositivo de destino 116 puede interactuar con un dispositivo externo de visualización, en lugar de incluir un dispositivo integrado de visualización.

[0028] El sistema 100 como se muestra en la Figura 1 es solo un ejemplo. En general, cualquier dispositivo de codificación y/o decodificación de vídeo digital puede realizar técnicas de codificación inter-intra. El dispositivo fuente 102 y el dispositivo de destino 116 son solo ejemplos de los dispositivos de codificación, en donde el dispositivo fuente 102 genera datos de video codificados para la 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 vídeo 200 y el decodificador de vídeo 300 representan ejemplos de dispositivos de codificación, en particular, un codificador de vídeo y un decodificador de vídeo, respectivamente. En algunos ejemplos, los dispositivos 102, 116 pueden operar de manera sustancialmente simétrica, de modo que cada uno de los dispositivos 102, 116 incluya componentes de codificación y decodificación de video. Por lo tanto, el sistema 100 puede permitir la transmisión de video unidireccional o bidireccional entre los dispositivos de video 102, 116, por ejemplo, para streaming de video, repetición de video, transmisión de video o videotelefonía.

[0029] En general, la fuente de video 104 representa una fuente de datos de video (es decir, datos de video sin procesar, no codificados) y proporciona una serie secuencial de imágenes (también denominadas "cuadros") de los datos de video al codificador de video 200, que codifica los 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 sin procesar capturado anteriormente y/o una interfaz de transmisión de video para recibir videos de un proveedor de contenidos 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 video en vivo, video archivado y video generado por computadora. En cada caso, el codificador de video 200 codifica los datos de video capturado, precapturado o generado por computadora. El codificador de vídeo 200 puede reorganizar las imágenes del orden recibido (a veces denominado "orden de visualización") en un orden de 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 emitir entonces los datos de vídeo codificados a través de la interfaz de salida 108 en un medio legible por ordenador 110 para su recepción

y/o recuperación mediante, por ejemplo, la interfaz de entrada 122 del dispositivo de destino 116.

[0030] La memoria 106 del dispositivo fuente 102 y la memoria 120 del dispositivo de destino 116 representan memorias de propósito general. En algún ejemplo, las memorias 106, 120 pueden almacenar datos de video sin procesar, por ejemplo, video sin procesar de la fuente de video 104 y datos de video sin procesar, decodificados del decodificador de video 300. Adicional o alternativamente, las memorias 106, 120 pueden almacenar instrucciones de software ejecutables por, por ejemplo, el codificador de video 200 y el decodificador de video 300, respectivamente. Aunque se muestran por separado del codificador de video 200 y el decodificador de video 300 en este ejemplo, debe entenderse que el codificador de video 200 y el decodificador de video 300 también pueden incluir memorias internas para propósitos funcionalmente similares o equivalentes. Además, las memorias 106, 120 pueden almacenar datos de video codificados, por ejemplo, la salida del codificador de video 200 y la entrada al decodificador de video 300. En algunos ejemplos, las porciones de las memorias 106, 120 pueden asignarse como una o más memorias intermedias de video, por ejemplo, para almacenar datos de video sin procesar, decodificados y/o codificados.

[0031] El medio legible por ordenador 110 puede representar cualquier tipo de medio o dispositivo capaz de transportar los datos de vídeo codificados desde el dispositivo fuente 102 al dispositivo de destino 116. En un ejemplo, el medio legible por ordenador 110 representa un medio de comunicación para permitir que el dispositivo fuente 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 ordenador. La interfaz de salida 108 puede modular una señal de transmisión que incluye los datos de vídeo codificados, y la interfaz de entrada 122 puede demodular la señal de transmisión recibida, según un estándar de comunicación, tal como un protocolo de comunicación inalámbrica. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o por cable, tales como una frecuencia de radio (RF) del espectro o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia, o una red global tal como Internet. El medio de comunicación puede incluir routers, conmutadores, estaciones base, o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo fuente 102 al dispositivo de destino 116.

[0032] En algunos ejemplos, el dispositivo fuente 102 puede enviar 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 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 el almacenamiento de datos de video codificado.

[0033] En algunos ejemplos, el dispositivo fuente 102 puede enviar 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 los datos de vídeo almacenados desde el servidor de archivos 114 mediante *streaming* o descarga. El dispositivo de destino 116 puede acceder a los datos de vídeo almacenados desde el servidor de archivos 114 mediante *streaming* o descarga. El servidor de archivos 114 puede representar un servidor web (por ejemplo, para un sitio web), un servidor de protocolo de transferencia de archivos (FTP), un dispositivo de red de entrega de contenido o un dispositivo de almacenamiento conectado a la red (NAS). El dispositivo de destino 116 puede acceder a los datos de video codificados del servidor de archivos 114 a través de cualquier conexión de datos estándar, lo que incluye 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, cable módem, etc.), o una combinación de ambos que es adecuada para acceder a los datos de video codificados y almacenados en el servidor de archivos 114. El servidor de archivos 114 y la interfaz de entrada 122 pueden configurarse para operar de acuerdo con un protocolo de *streaming* de flujo continuo, un protocolo de transmisión de descarga o una combinación de estos.

[0034] La interfaz de salida 108 y la interfaz de entrada 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 los ejemplos en los que la 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, como datos de vídeo codificados, de acuerdo con un estándar de comunicación celular, como 4G, 4G-

LTE (Evolución a largo plazo), LTE Avanzada, 5G o similares. En algunos ejemplos en los que 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, como datos de vídeo codificados, de acuerdo con otros estándares inalámbricos, como una memoria descriptiva IEEE 802.11, una memoria descriptiva 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 respectivos dispositivos de sistema en un chip (SoC). Por ejemplo, el dispositivo fuente 102 puede incluir un dispositivo SoC para realizar la funcionalidad atribuida al codificador de video 200 y/o 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 interfaz de entrada 122.

[0035] Las técnicas de la presente divulgación se pueden aplicar a la codificación de video para diversas aplicaciones multimedia, tales como transmisiones por televisión inalámbrica, transmisiones por televisión por cable, transmisiones por televisión satelital, transmisiones de video por streaming en Internet, tales como streaming adaptativo dinámico en 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.

[0036] La interfaz de entrada 122 del dispositivo de destino 116 recibe un flujo de bits de vídeo codificado desde un medio legible por ordenador 110 (por ejemplo, el dispositivo de almacenamiento 112, el servidor de archivos 114 o similar). El flujo de bits de video codificado puede incluir información de señalización definida por el codificador de video 200, que también es utilizada por el decodificador de video 300, como elementos de sintaxis que tienen valores que describen características y/o procesamiento de bloques de video u 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 vídeo decodificados a un usuario. El dispositivo de visualización 118 puede representar cualquiera de los diversos dispositivos de visualización, tales como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodo orgánico de emisión de luz (OLED), u otro tipo de dispositivo de visualización.

[0037] Aunque no se muestra en la Figura 1, en algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden cada uno ser integrados con un codificador de audio y/o un decodificador de audio, y pueden incluir unidades MUX-

DEMUX apropiadas, u otro hardware y/o software, para manejar flujos múltiples, incluida la codificación de audio y video en un flujo común de datos. Si correspondiera, las unidades MUX-DEMUX pueden adecuarse al protocolo de multiplexor ITU H.223, o a otros protocolos, tales como el protocolo de datagrama de usuario (UDP).

[0038] El codificador de video 200 y el decodificador de video 300 pueden implementarse como cualquiera de una variedad de circuito codificador y/o decodificador adecuado, tal como uno o más microprocesadores, procesadores de señal digital (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas programables en campo (FPGA), circuitos discretos lógicos, software, hardware, firmware o cualquier combinación de estos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar las instrucciones para el software en un medio legible por ordenador adecuado, no transitorio, y ejecutar las instrucciones en hardware mediante el uso de uno o más procesadores para llevar a cabo las técnicas de esta divulgación. Cada uno de codificador de video 200 y decodificador de video 300 pueden incluirse en uno o más codificadores o decodificadores, cualquiera de los cuales pueden ser integrados como parte de un codificador/decodificador combinado (CODEC) en un dispositivo respectivo. Un dispositivo que incluye un codificador de video 200 y/o un decodificador de video 300 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

[0039] El codificador de video 200 y el decodificador de video 300 pueden operar de acuerdo con un estándar de codificación de video, como ITU-T H.265, también conocido como codificación de video de alta eficiencia (HEVC) o extensiones de este, como las extensiones de codificación de vídeo multivista y/o escalable. Alternativamente, el codificador de video 200 y el decodificador de video 300 pueden operar de acuerdo con otros estándares de propiedad o de la industria, como el modelo de prueba de exploración conjunta (JEM) o ITU-T H.266, también denominado codificación de video versátil (VVC). Las técnicas de esta divulgación, sin embargo, no se limitan a ningún estándar de codificación particular.

[0040] En general, el codificador de vídeo 200 y el decodificador de vídeo 300 pueden realizar una codificación de imágenes basada en bloques. El término "bloque" generalmente se refiere a una estructura que incluye datos para ser procesados (por ejemplo, codificados, decodificados o usados de otro modo en el proceso de codificación y/o decodificación). Por ejemplo, un bloque puede incluir una matriz bidimensional de muestras de datos de luminancia y/o 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 de rojo, verde y azul (RGB) para muestras de una imagen, el codificador de video 200 y el decodificador de video 300 pueden codificar componentes de luminancia y crominancia, en donde los componentes de crominancia pueden incluir componentes de crominancia de tono rojo y 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.

[0041] 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 tanto, las referencias a la codificación de una imagen o un bloque deben entenderse generalmente como valores de codificación para los elementos de sintaxis que forman la imagen o el bloque.

[0042] HEVC define varios bloques, incluidas las unidades de codificación (CU), las unidades de predicción (PU) y las unidades de transformación (TU). 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 cuaternario. Es decir, el codificador de video divide las CTU y CU en cuatro cuadrados iguales que no se superponen, y cada nodo de árbol cuaternario 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 dividir aún más PU y TU. Por ejemplo, en HEVC, un árbol cuaternario residual (RQT) representa la partición de TU. En HEVC, las PU representan datos de interpredicción, mientras que las TU representan datos residuales. Las CU que son intrapredichas incluyen información intrapredicción, como una indicación intramodo.

[0043] Como otro ejemplo, el codificador de video 200 y el decodificador de video 300

pueden configurarse para funcionar de acuerdo con JEM o VVC. De acuerdo con JEM o 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 dividir una CTU de acuerdo con una estructura de árbol, tal como una estructura de árbol binario de árbol cuaternario (QTBT) o una estructura de árbol de tipos múltiples (MTT). 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 dividido de acuerdo con la partición de árbol cuaternario y un segundo nivel dividido 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).

[0044] En una estructura de partición MTT, los bloques se pueden particionar mediante el uso de una partición de árbol cuaternario (QT), una partición de árbol binario (BT) y uno o más tipos de particiones de árbol triple (TT). Una partición de árbol triple es una partición en la que un bloque se divide en tres subbloques. En algunos ejemplos, una partición de árbol triple divide un bloque en tres subbloques sin dividir el bloque original por el centro. Los tipos de particiones en MTT (por ejemplo, QT, BT y TT) pueden ser simétricos o asimétricos.

[0045] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden usar una sola 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, 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).

[0046] El codificador de video 200 y el decodificador de video 300 pueden configurarse para usar particiones de cuatro cuaternario por HEVC, particiones QTBT, particiones MTT u otras estructuras de particiones. Para fines de explicación, la descripción de las técnicas de esta divulgación se presenta con respecto a la partición QTBT. Sin embargo, debe entenderse que las técnicas de esta divulgación también se pueden aplicar a codificadores de video configurados para usar particiones de árbol cuaternario, u otros tipos de particiones también.

[0047] Esta divulgación puede usar "NxN" y "N por N" indistintamente para referirse a las dimensiones de muestra de un bloque (como un CU u otro bloque de video) en términos de dimensiones verticales y horizontales, por ejemplo, muestras de 16x16 o

muestras de 16 por 16. En general, una CU de 16x16 tendrá 16 muestras en dirección vertical ($y = 16$) y 16 muestras en dirección horizontal ($x = 16$). Asimismo, 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 disponerse en filas y columnas. Además, las CU no necesitan tener necesariamente la misma cantidad de muestras en la dirección horizontal que en la dirección vertical. Por ejemplo, las CU pueden comprender muestras NxM, en donde M no es necesariamente igual a N.

[0048] El codificador de video 200 codifica datos de video para CU que representan predicción y/o información residual y otra información. La información de predicción indica cómo se va a predecir la CU para formar un bloque de predicción para la CU. La información residual generalmente representa diferencias muestra por muestra entre muestras de la CU antes de la codificación y el bloque de predicción.

[0049] Para predecir una CU, el codificador de vídeo 200 puede formar generalmente un bloque de predicción para la CU a través de 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 vídeo 200 puede generar el bloque de predicción mediante el uso de uno o más vectores de movimiento. El codificador de video 200 generalmente puede realizar una búsqueda de movimiento para identificar un bloque de referencia que se asemeje mucho a 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 mediante el uso de una suma de diferencia absoluta (SAD), suma de diferencias cuadradas (SSD), diferencia absoluta media (MAD), diferencias cuadráticas medias (MSD) u otros cálculos de diferencias similares para determinar si una referencia de bloque se asemeja mucho a la CU actual. En algunos ejemplos, el codificador de video 200 puede predecir la CU actual mediante el uso de predicción unidireccional o predicción bidireccional.

[0050] Algunos ejemplos de JEM y 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 vídeo 200 puede determinar dos o más vectores de movimiento que representan un movimiento no traslacional, como acercar o alejar, rotación, movimiento en perspectiva u otros tipos de movimiento irregular.

[0051] Para realizar la intrapredicción, el codificador de vídeo 200 puede seleccionar un modo de intrapredicción para generar el bloque de predicción. Algunos ejemplos de JEM y 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) desde el cual predecir muestras del bloque actual. Por lo general, dichas muestras pueden estar arriba, arriba y a la izquierda, o a la izquierda del bloque actual en la misma imagen que el bloque actual, asumiendo que el codificador de video 200 codifica CTU y CU en orden de exploración de trama (de izquierda a derecha, de arriba a abajo).

[0052] De acuerdo con las técnicas de esta divulgación, el codificador de video 200 puede realizar una predicción inter-intra de bloques de datos de vídeo (por ejemplo, CU). Es decir, el codificador de vídeo 200 puede formar un bloque de predicción mediante la utilización de tanto un bloque de interpredicción como un bloque de intrapredicción. El codificador de video 200 puede formar el bloque de predicción final para un bloque actual al realizar una combinación ponderada de muestras del bloque de interpredicción y el bloque de intrapredicción. Cuando un bloque actual es un bloque de crominancia, el codificador de video 200 puede determinar las ponderaciones de acuerdo con un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que está codificado por intrapredicción (y/o un número de bloques que están codificados por interpredicción). Es decir, en lugar de determinar las ponderaciones de acuerdo con los bloques vecinos al bloque de crominancia actual, el codificador de video 200 puede determinar el número de bloques intrapredichos o interpredichos vecinos del bloque de luminancia correspondiente al bloque de crominancia actual, luego determinar las ponderaciones en función del número de bloques intrapredichos o interpredichos vecinos al bloque de luminancia.

[0053] Como ejemplo, el codificador de video 200 puede determinar si un bloque vecino por encima y/o un bloque vecino por la izquierda al bloque de luminancia correspondiente están interpredichos. Cuando ninguno de estos bloques vecinos está interpredicho, el codificador de video 200 puede determinar una ponderancia de 3 que se aplicará al bloque de intrapredicción y una ponderancia de 1 que se aplicará al bloque de interpredicción. Cuando ambos de estos bloques vecinos está interpredicho, el codificador de video 200 puede determinar una ponderancia de 3 que se aplicará al bloque de interpredicción y una ponderancia de 1 que se aplicará al bloque de

intrapredicción. Cuando uno de estos bloques vecinos está interpredicho y el otro está intrapredicho, el codificador de video 200 puede determinar ponderaciones de 2 que se aplicarán tanto al bloque de interpredicción como al bloque de intrapredicción. En algunos ejemplos, el codificador de video 200 puede contar bloques vecinos predichos mediante el uso de predicción inter-intra y/o copia intrabloque como bloques codificados por interpredicción. El decodificador de vídeo 300 puede realizar sustancialmente el mismo proceso para determinar las ponderaciones que se aplicarán a los bloques de interpredicción e intrapredicción para un bloque de crominancia actual en función de los modos de predicción para bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual.

[0054] 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 vídeo 200 puede codificar datos que representen cuál de los diversos modos de interpredicción disponibles se utiliza, así como información de movimiento para el modo correspondiente. Para la interpredicción unidireccional o bidireccional, por ejemplo, el codificador de vídeo 200 puede codificar vectores de movimiento mediante el uso de la predicción avanzada de vectores de movimiento (AMVP) o el modo de fusión. El codificador de vídeo 200 puede utilizar modos similares para codificar vectores de movimiento para el modo de compensación de movimiento afín.

[0055] Después de la predicción, tal como intrapredicción o interpredicción de un bloque, el codificador de video 200 puede calcular datos residuales para el bloque. Los datos residuales, como un bloque residual, representan las diferencias muestra por muestra entre el bloque y un bloque de predicción para el bloque, formado mediante el uso del modo de predicción correspondiente. El codificador de vídeo 200 puede aplicar una o más transformaciones al bloque residual, para producir datos transformados en un dominio de transformación en lugar del dominio de muestra. Por ejemplo, el codificador de video 200 puede aplicar una transformación de coseno discreta (DCT), una transformación de entero, una transformación de ondículas o una transformación conceptualmente similar a los datos de video residuales. Además, el codificador de vídeo 200 puede aplicar una transformación secundaria después de la primera transformación, como una transformación secundaria no separable dependiente del modo (MDNSST), una transformación dependiente de la señal, una transformación de Karhunen-Loeve (KLT) o similares. El codificador de vídeo 200 produce coeficientes de transformación tras la aplicación de una o más transformaciones.

[0056] Como se indicó anteriormente, siguiendo cualquier transformación para

producir coeficientes de transformación, el codificador de video 200 puede realizar la cuantificación de los coeficientes de transformación. Cuantificación refiere generalmente a un proceso donde los coeficientes de transformada se cuantifican para reducir posiblemente el número de datos utilizados para representar los coeficientes, lo que proporciona una compresión adicional. Al realizar el proceso de cuantificación, el codificador de vídeo 200 puede reducir la profundidad de bits asociada con algunos o todos los coeficientes. 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 puede realizar un desplazamiento a la derecha bit a bit del valor que se va a cuantificar.

[0057] Después de la cuantificación, el codificador de video 200 puede explorar los coeficientes de transformación, lo que produce un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformación cuantificados. La exploración puede diseñarse para colocar coeficientes de energía más alta (y por lo tanto de frecuencia más baja) al frente del vector y para colocar coeficientes de transformación de energía más baja (y por lo tanto de frecuencia más alta) en la parte posterior del vector. En algunos ejemplos, el codificador de video 200 puede utilizar un orden de exploración predefinido para explorar los coeficientes de transformación cuantificados para producir un vector serializado, y luego codificar por entropía los coeficientes de transformación cuantificados del vector. En otros ejemplos, el codificador de video 200 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformación 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 su uso por el decodificador de video 300 en la decodificación de los datos de video.

[0058] Para realizar CABAC, el codificador de vídeo 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 tienen valor cero o no. La determinación de la probabilidad puede ser en función de un contexto asignado al símbolo.

[0059] El codificador de video 200 puede generar además datos de sintaxis, como datos de sintaxis basados en bloques, datos de sintaxis basados en imágenes y datos

de sintaxis basados en secuencias, al decodificador de video 300, por ejemplo, en un encabezado de imagen, un encabezado de bloque, un encabezado de segmento, u otros datos de sintaxis, como un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS) o un conjunto de parámetros de vídeo (VPS). El decodificador de vídeo 300 puede decodificar igualmente dichos datos de sintaxis para determinar cómo decodificar los datos de vídeo correspondientes.

[0060] 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. Finalmente, el decodificador de video 300 puede recibir el flujo de bits y decodificar los datos de video codificados.

[0061] 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 mediante el uso de CABAC de una manera sustancialmente similar, aunque recíproca al proceso de codificación CABAC del codificador de video 200. Los elementos de sintaxis pueden definir la información de 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 además la predicción y la información residual para bloques (por ejemplo, CU) de datos de vídeo.

[0062] La información residual puede representarse, por ejemplo, mediante coeficientes de transformación cuantificados. El decodificador de video 300 puede cuantificar y transformar de forma inversa los coeficientes de transformación cuantificados de un bloque para reproducir un bloque residual para el bloque. El decodificador de video 300 utiliza un modo de predicción señalado (intrapredicción 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 combinar entonces 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, como realizar un proceso de desbloqueo para reducir los artefactos visuales a lo largo de los límites del bloque.

[0063] De acuerdo con las técnicas de esta divulgación, el codificador de video 200

y/o el decodificador de video 300 pueden configurarse para realizar codificación inter-intra. Es decir, el codificador de video 200 y/o el decodificador de video 300 pueden predecir un bloque de datos de video mediante el uso de predicción inter-intra combinada de acuerdo con cualquiera o todas de las técnicas descritas en la presente.

[0064] Por ejemplo, el codificador de video 200 y el decodificador de video 300 pueden determinar de forma adaptativa las ponderaciones para aplicar a las muestras de una interpredicción y una intrapredicción en función de un esquema de ponderación independiente de la posición. Por ejemplo, el codificador de video 200 y el decodificador de video 300 pueden determinar de forma adaptativa las ponderaciones de acuerdo con (por ejemplo, como una función de) la información de codificación, como el número de bloques vecinos intracodificados y/o intercodificados, el número de interpredicciones de los bloques indexados de fusión (unipredicción o bipredicción), el tamaño del bloque actual. Sea (w_{Inter} , w_{Intra}) las ponderaciones para las muestras de predicción inter e intra en la descripción siguiente. Es decir, w_{Inter} representa un valor de ponderación aplicado a muestras de un bloque de interpredicción y w_{Intra} representa un valor de ponderación aplicado a muestras de un bloque de intrapredicción. En algunos ejemplos, $w_{\text{Inter}} + w_{\text{Intra}} = 1$, donde w_{Inter} y w_{Intra} son valores racionales entre 0 y 1.

[0065] Esta divulgación en general 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 señaló 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, 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.

[0066] Las Figuras 2A y 2B son diagramas conceptuales que ilustran un ejemplo de estructura de árbol binario de árbol cuaternario (QTBT) 130 y una unidad de árbol de codificación (CTU) 132 correspondiente. Las líneas continuas representan la división de árbol cuaternario y las líneas de puntos indican la división de árbol binario. En cada nodo dividido (es decir, no hoja) del árbol binario, se señala una bandera para indicar qué tipo de división (es decir, horizontal o vertical) se utiliza, donde 0 indica división horizontal y 1 indica división vertical en este ejemplo. Para la división de árbol

cuaternario, no es necesario indicar el tipo de división, ya que los nodos de árbol cuaternario dividen un bloque horizontal y verticalmente en 4 subbloques con el mismo tamaño. En consecuencia, el codificador de video 200 puede codificar, y el decodificador de video 300 puede decodificar, elementos de sintaxis (como dividir información) para un nivel de árbol de región de la estructura QTBT 130 (es decir, las líneas continuas) y elementos de sintaxis (como dividir información) para un nivel de árbol de predicción de la estructura QTBT 130 (es decir, las líneas discontinuas). El codificador de video 200 puede codificar, y el decodificador de video 300 puede decodificar, datos de video, tales como datos de predicción y transformación, para CU representadas por nodos hoja terminal de la estructura QTBT 130.

[0067] En general, la CTU 132 de la Figura 2B puede estar asociada con parámetros que definen tamaños de bloques correspondientes a nodos de la estructura QTBT 130 en el primer y segundo nivel. Estos parámetros pueden incluir un tamaño de CTU (que representa un tamaño de CTU 132 en muestras), un tamaño mínimo de árbol cuaternario (MinQTSIZE, que representa un tamaño mínimo de nodo hoja de árbol cuaternario permitido), un tamaño máximo de árbol binario (MaxBTSIZE, que representa un tamaño de nodo raíz de árbol binario máximo permitido), una profundidad máxima de árbol binario (MaxBTDepth, que representa una profundidad máxima permitida de árbol binario) y un tamaño mínimo de árbol binario (MinBTSIZE, que representa el tamaño mínimo permitido de nodo hoja de árbol binario).

[0068] El nodo raíz de una estructura QTBT correspondiente a una CTU puede tener cuatro nodos secundarios en el primer nivel de la estructura QTBT, cada uno de los cuales puede particionarse de acuerdo con la partición de árbol cuaternario. Es decir, los nodos del primer nivel son nodos hoja (que no tienen nodos secundarios) o tienen cuatro nodos secundarios. El ejemplo de la estructura QTBT 130 representa tales nodos que incluyen el nodo principal y los nodos secundarios que tienen líneas continuas para las ramas. Si los nodos del primer nivel no son más grandes que el tamaño máximo permitido del nodo raíz del árbol binario (MaxBTSIZE), se pueden dividir aún más mediante los respectivos árboles binarios. La división del árbol binario de un nodo se puede iterar hasta que los nodos resultantes de la división alcancen el tamaño mínimo permitido del nodo hoja del árbol binario (MinBTSIZE) o la profundidad máxima permitida del árbol binario (MaxBTDepth). El ejemplo de la estructura QTBT 130 representa tales nodos que tienen líneas discontinuas para las ramas. El nodo hoja del árbol binario se denomina unidad de codificación (CU), que se utiliza para la predicción (por ejemplo, predicción intraimagen o interimagen) y transformación, sin

más particiones. Como se analizó anteriormente, las CU también se pueden denominar "bloques de video" o "bloques".

[0069] En un ejemplo de la estructura de partición QTBT, el tamaño de CTU se establece en 128x128 (muestras de luminancia y dos muestras de crominancia de 64x64 correspondientes), MinQTSIZE se establece en 16x16, MaxBTSIZE se establece en 64x64, MinBTSIZE (para ancho y altura) se establece en 4 y MaxBTDepth se establece en 4. La partición del árbol cuaternario se aplica primero a la CTU para generar nodos hoja de árbol cuaternario. Los nodos hoja de árbol cuaternario pueden tener un tamaño de 16x16 (es decir, MinQTSIZE) a 128x128 (es decir, el tamaño de CTU). Si el nodo hoja de árbol cuaternario es 128x128, el árbol binario no lo dividirá más, ya que el tamaño excede el MaxBTSIZE (es decir, 64x64, en este ejemplo). De lo contrario, el árbol binario dividirá aún más el nodo hoja de árbol cuaternario. Por lo tanto, el nodo hoja de árbol cuaternario es también el nodo raíz del árbol binario y tiene la profundidad del árbol binario como 0. Cuando la profundidad del árbol binario alcanza MaxBTDepth (4, en este ejemplo), no se permite más división. Cuando el nodo de árbol binario tiene un ancho igual a MinBTSIZE (4, en este ejemplo), implica que no se permite más división horizontal. De manera similar, un nodo de árbol binario que tiene una altura igual a MinBTSIZE implica que no se permite más división vertical para ese nodo de árbol binario. Como se señaló anteriormente, los nodos hoja de árbol binario se denominan CU y se procesan adicionalmente de acuerdo con la predicción y se transforman sin más particiones.

[0070] Las Figuras 3A-3F son diagramas conceptuales que ilustran ubicaciones de ejemplo de bloques vecinos a bloques actuales. En particular, la Figura 3A representa un ejemplo del bloque actual 140 que tiene el bloque vecino superior 142 y el bloque vecino izquierdo 144, la Figura 3B representa un ejemplo del bloque actual 146 que tiene el bloque vecino superior derecho 148 y el bloque vecino inferior izquierdo 150, la Figura 3C representa un ejemplo del bloque actual 152 que tiene el bloque vecino superior 154, el bloque vecino izquierdo 156 y el bloque vecino superior izquierdo (superior-i) 158, la Figura 3D representa un ejemplo de bloque actual 160 que tiene el bloque vecino superior derecho 162, el bloque vecino inferior izquierdo (inferior-i) 164 y el bloque vecino superior izquierdo 166, la Figura 3E representa un ejemplo del bloque actual 168 que tiene el bloque vecino superior 170, el bloque vecino superior derecho 172, el bloque vecino izquierdo 174 y el bloque vecino inferior izquierdo 176, la Figura 3F representa un ejemplo del bloque actual 178 que tiene el bloque vecino superior 180, el bloque vecino superior derecho 182, el bloque vecino izquierdo 184, el bloque

vecino inferior izquierdo 186 y el bloque vecino superior izquierdo 188.

[0071] De acuerdo con las técnicas de esta divulgación, el codificador de video 200 y el decodificador de video 300 pueden determinar de manera adaptativa las ponderaciones para intra e intermuestras en función de una serie de bloques vecinos intra y/o intercodificados, por ejemplo, como se describe en la solicitud estadounidense N.º 16/684.379, presentada el 14 de noviembre de 2019, cuyo contenido completo se incorpora en la presente por referencia. Sin embargo, de acuerdo con las técnicas de esta divulgación, el codificador de video 200 y el decodificador de video 300 pueden combinar solamente muestras de interpredicción e intapredicción, sin combinación de intrapredicción dependiente de la posición (PDPC). El codificador de video 200 y el decodificador de video 300 pueden usar bloques vecinos de referencia para determinar ponderaciones, en donde los bloques vecinos de referencia pueden ser cualquier combinación de bloques vecinos superior, superior derecho, superior izquierdo, izquierdo o inferior izquierdo, por ejemplo, como se muestra en los diversos ejemplos de las Figuras 3A–3F.

[0072] En un ejemplo, por ejemplo, como se muestra en la Figura 3A, el codificador de vídeo 200 y el decodificador de vídeo 300 utilizan el bloque vecino superior 142 y el bloque vecino izquierdo 144 para determinar las ponderaciones a aplicar a las muestras de intrapredicción e interpredicción para la predicción inter-intra del bloque actual 140. Es decir, el bloque vecino superior 142 y el bloque vecino izquierdo 144 se utilizan para la verificación intracodificada de vecinos para el bloque actual 140, en este ejemplo.

[0073] En otro ejemplo, por ejemplo, como se muestra en la Figura 3B, el codificador de vídeo 200 y el decodificador de vídeo 300 utilizan el bloque vecino superior derecho 148 y el bloque vecino inferior izquierdo 150 para determinar las ponderaciones a aplicar a las muestras de intrapredicción e interpredicción para la predicción inter-intra del bloque actual 146. Es decir, el bloque vecino superior derecho 148 y el bloque vecino inferior izquierdo 150 se utilizan para la verificación intracodificada de vecinos para el bloque actual 146, en este ejemplo.

[0074] En otro ejemplo, por ejemplo, como se muestra en la Figura 3C, el bloque vecino superior 154, el bloque vecino izquierdo 156 y el bloque vecino superior izquierdo 158 se utilizan para la verificación de vecinos intracodificados para el bloque actual 152.

[0075] En otro ejemplo, por ejemplo, como se muestra en la Figura 3D, el bloque vecino superior derecho 162, el bloque vecino inferior izquierdo 164 y el bloque vecino

superior izquierdo 166 se utilizan para la verificación de vecinos intracodificados para el bloque 160 actual.

[0076] En otro ejemplo, por ejemplo, como se muestra en la Figura 3E, el bloque vecino superior 170, el bloque vecino superior derecho 172, el bloque vecino izquierdo 174 y el bloque vecino inferior izquierdo 176 se utilizan para la verificación de vecinos intracodificados para el bloque actual 168.

[0077] En otro ejemplo, por ejemplo, como se muestra en la Figura 3F, el bloque vecino superior 180, el bloque vecino superior derecho 182, el bloque vecino izquierdo 184, el bloque vecino inferior izquierdo 186 y el bloque vecino superior izquierdo 188 se utilizan para la verificación de vecinos intracodificados para el bloque actual 178.

[0078] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 seleccionan bloques vecinos de referencia de acuerdo con el tamaño de bloque del bloque actual (por ejemplo, blkWidth y blkHeight). Por ejemplo, si blkWidth y blkHeight son idénticos, el codificador de vídeo 200 y el decodificador de vídeo 300 pueden utilizar la combinación de la Figura 3D, la Figura 3E, o la Figura 3F. En otro ejemplo, si blkWidth y blkHeight son diferentes, el codificador de video 200 y el decodificador de video 300 pueden usar superior derecho e inferior izquierdo, por ejemplo, como se muestra en la Figura 3B.

[0079] En otro ejemplo, si blkWidth y blkHeight son idénticos, entonces el codificador de video 200 y el decodificador de video 300 pueden seleccionar las posiciones del bloque vecino simétricamente a lo largo de las dimensiones de ancho y altura del bloque actual, por ejemplo, de acuerdo con los ejemplos de las Figuras 3A–3F. Por otro lado, si blkWidth y blkHeight son diferentes, el codificador de video 200 y el decodificador de video 300 pueden seleccionar las posiciones de bloques vecinos asimétricamente a lo largo de las dimensiones de ancho y altura del bloque actual. Por ejemplo, si blkWidth es mayor que blkHeight, el codificador de video 200 y el decodificador de video 300 pueden seleccionar los bloques vecinos como superior derecho e izquierdo, mientras que si blkWidth es menor que blkHeight, el codificador de video 200 y el decodificador de video 300 pueden seleccionar los bloques vecinos como siendo inferior izquierdo y superior.

[0080] En algunos ejemplos, si el bloque vecino verificado está codificado por IBC/CPR (copia intrabloque/referenciación de imagen actual), el bloque vecino puede considerarse como un bloque intercodificado.

[0081] En algunos ejemplos, si el bloque vecino verificado está codificado por IBC/CPR, el bloque vecino puede considerarse como un bloque intracodificado.

[0082] En algunos ejemplos, cuando un bloque vecino verificado es un bloque intra-inter combinado, el bloque vecino puede considerarse como un bloque intercodificado.

[0083] En algunos ejemplos, cuando un bloque vecino verificado es un bloque intra-inter combinado, el bloque vecino puede considerarse como un bloque intracodificado.

[0084] En algunos ejemplos, cuando se habilita un único árbol de codificación luminancia-crominancia o árboles de codificación luminancia-crominancia duales (separados), el codificador de video 200 y el decodificador de video 300 pueden determinar las ponderaciones para la combinación de los bloques de crominancia de acuerdo con la verificación intra del bloque luminancia correspondiente o bloques vecinos del bloque luminancia correspondiente.

[0085] En algunos ejemplos, cuando se habilita la codificación de árbol dual, el codificador de video 200 y el decodificador de video 300 pueden determinar las ponderaciones para la combinación de los bloques de crominancia de acuerdo con la verificación intra de los bloques de crominancia vecinos.

[0086] El codificador de video 200 y el decodificador de video 300 pueden determinar las ponderaciones en función de una verificación intra. Sea w_{Inter} y w_{Intra} las ponderaciones de las muestras inter e intra en la mezcla inter-intra. Las ponderaciones se pueden normalizar por 2^n siendo n un número entero, que es igual a la suma de las ponderaciones. En otras palabras, estas ponderaciones se pueden normalizar en 4, 8, 16,..., lo que se puede implementar con una simple operación de cambio a la derecha.

[0087] En algunos ejemplos, si todos los bloques vecinos verificados son intracodificados, el codificador de video 200 y el decodificador de video 300 pueden determinar que la ponderación de las muestras intra es mayor que el de las muestras inter (por ejemplo, $(w_{Inter}, w_{Intra}) = (1, 3)$ o $(w_{Inter}, w_{Intra}) = (3, 5)$).

[0088] En algunos ejemplos, si todos los bloques vecinos verificados no son intracodificados, el codificador de video 200 y el decodificador de video 300 pueden determinar que la ponderación de las muestras intra es menor que el de las muestras inter (por ejemplo, $(w_{Inter}, w_{Intra}) = (3, 1)$ o $(w_{Inter}, w_{Intra}) = (5, 3)$).

[0089] En algunos ejemplos, si sólo uno de los bloques vecinos verificados está intracodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar que las ponderaciones para muestras intra e inter son idénticos.

[0090] En algunos ejemplos, si los bloques vecinos superior izquierdo, superior derecho e inferior izquierdo están todos intracodificados, el codificador de video 200 y el decodificador de video 300 pueden determinar que las ponderaciones (w_{Inter}, w_{Intra}) son $(1, 3)$ o $(3, 5)$. En otro ejemplo, si al menos uno de los bloques vecinos

superior izquierdo, superior derecho o inferior izquierdo está intracodificado, el codificador de video 200 y el decodificador de video 300 pueden establecer (wInter, wIntra) igual a (2, 2) o (4, 4). En aún otro ejemplo, si todos los bloques vecinos superior izquierdo, superior derecho e inferior izquierdo no están intracodificados, el codificador de video 200 y el decodificador de video 300 pueden establecer (wInter, wIntra) igual a (3, 1) o (5, 3).

[0091] En algunos ejemplos, si al menos uno de los bloques verificados es un bloque codificado en MHI (intra de hipótesis múltiples), el codificador de vídeo 200 y el decodificador de vídeo 300 pueden copiar la ponderación del bloque actual a partir de la ponderación de uno de los bloques verificados.

[0092] En algunos ejemplos, si solo uno de estos bloques verificados es intracodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar que la ponderación de las muestras intra es mayor que el de las muestras inter (por ejemplo, (WInter, wIntra) = (1, 3) o (wInter, wIntra) = (3, 5)).

[0093] En algunos ejemplos, si solo uno de estos bloques verificados es intracodificado, el codificador de video 200 y el decodificador de video 300 pueden determinar que la ponderación de las muestras intra es menor que el de las muestras inter (por ejemplo, (WInter, wIntra) = (3, 1) o (wInter, wIntra) = (5, 3)).

[0094] En algunos ejemplos, si todos estos bloques verificados no son intracodificados, el codificador de video 200 y el decodificador de video 300 pueden determinar que la ponderación de las muestras intra es mayor que el de las muestras inter (por ejemplo, (WInter, wIntra) = (1, 3) o (wInter, wIntra) = (3, 5)).

[0095] En algunos ejemplos, el codificador de vídeo 200 y el decodificador de vídeo 300 pueden determinar (wInter, wIntra) de acuerdo con el modo de intrapredicción de los bloques vecinos verificados. En un ejemplo, si hay al menos un bloque vecino que se codifica mediante el uso del modo intra DC o el modo planar, la ponderación de las muestras intra puede ser mayor que el de las muestras inter (por ejemplo, (WInter, wIntra) = (1, 3) o (wInter, wIntra) = (3, 5)).

[0096] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden determinar (wInter, wIntra) en función del número de interpretaciones del bloque indexado de fusión, es decir, si el bloque de índice de fusión se predice mediante el uso de una predicción unidireccional o bidireccional.

[0097] En algunos ejemplos, si el bloque indexado de fusión es de bipredicción, el codificador de video 200 y el decodificador de video 300 pueden establecer wInter mayor que wIntra (por ejemplo, (wInter, wIntra) = (3, 1) o (wInter, wIntra) = (5, 3)).

[0098] En algunos ejemplos, si el bloque indexado de fusión es de unipredicción, el codificador de video 200 y el decodificador de video 300 pueden establecer w_{lnter} mayor que w_{lntra} (por ejemplo, $(w_{lnter}, w_{lntra}) = (3, 1)$ o $(w_{lnter}, w_{lntra}) = (5, 3)$).

[0099] En algunos ejemplos, si el bloque indexado de fusión es de bipredicción, el codificador de video 200 y el decodificador de video 300 pueden establecer w_{lnter} mayor que w_{lntra} (por ejemplo, $(w_{lnter}, w_{lntra}) = (3, 1)$ o $(w_{lnter}, w_{lntra}) = (5, 3)$). Mientras tanto, para un bloque indexado de fusión con unipredicción, el codificador de video 200 y el decodificador de video 300 pueden determinar (w_{lnter}, w_{lntra}) mediante el uso del número de intra y/o interbloques vecinos, por ejemplo, como se analizó anteriormente.

[0100] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden determinar (w_{lnter}, w_{lntra}) de acuerdo con el tamaño del bloque actual. El tamaño ($SIZE_{blk}$) del bloque actual es en función de su ancho y altura.

[0101] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden determinar el tamaño del bloque actual como el valor mínimo de ancho y altura del bloque actual. En otro ejemplo, el codificador de video 200 y el decodificador de video 300 pueden determinar el tamaño del bloque actual como el valor máximo de ancho y altura del bloque actual. En otro ejemplo más, el codificador de video 200 y el decodificador de video 300 pueden determinar el tamaño del bloque actual mediante la multiplicación (es decir, el producto) del ancho y altura del bloque actual. En aun otro ejemplo, el codificador de video 200 y el decodificador de video 300 pueden determinar el tamaño del bloque actual mediante la suma del ancho y altura del bloque actual.

[0102] En algunos ejemplos, si $SIZE_{blk}$ es mayor que un umbral predefinido, el codificador de video 200 y el decodificador de video 300 pueden establecer w_{lnter} mayor o menor que w_{lntra} (por ejemplo, $(w_{lnter}, w_{lntra}) = (3, 1)$ o $(w_{lnter}, w_{lntra}) = (5, 3)$).

[0103] En algunos ejemplos, si $SIZE_{blk}$ es menor que un umbral predefinido, el codificador de video 200 y el decodificador de video 300 pueden establecer w_{lnter} mayor o menor que w_{lntra} (por ejemplo, $(w_{lnter}, w_{lntra}) = (3, 1)$ o $(w_{lnter}, w_{lntra}) = (5, 3)$).

[0104] En algunos ejemplos, si el bloque indexado de fusión es de bipredicción, el codificador de video 200 y el decodificador de video 300 pueden establecer w_{lnter} mayor o menor que w_{lntra} (por ejemplo, $(w_{lnter}, w_{lntra}) = (3, 1)$ o $(w_{lnter}, w_{lntra}) = (5, 3)$). Mientras tanto, para el bloque indexado de fusión con unipredicción, el codificador

de video 200 y el decodificador de video 300 pueden determinar (w_{Inter} , w_{Intra}) adaptativamente de acuerdo con el tamaño del bloque actual.

[0105] En algunos ejemplos, un conjunto de pares de ponderaciones (w_{Inter} , w_{Intra}) puede estar predefinido. El conjunto de pares de ponderaciones puede indicarse en el encabezado del flujo de bits o en el conjunto de parámetros de secuencia (SPS). Es decir, el codificador de video 200 puede codificar el conjunto de pares de ponderaciones, mientras que el decodificador de video 300 puede decodificar el conjunto de pares de ponderaciones. Durante el proceso de codificación de un bloque MHI, el codificador de video 200 puede determinar la mejor ponderación mediante el uso de la evaluación de la distorsión de la velocidad (RD). En este ejemplo, el codificador de video 200 puede derivar costos RD de todos los elementos en el par de ponderaciones. El codificador de video 200 puede seleccionar el par que tenga el costo RD mínimo para codificar el bloque actual y señalar el índice de este par en el flujo de bits. El codificador de video 200 puede codificar el índice mediante un algoritmo CABAC mediante el uso de bypass o codificación por entropía basada en el contexto. El decodificador de video 300 puede realizar un proceso de decodificación para obtener el índice para determinar las ponderaciones (por ejemplo, el par de ponderaciones) a aplicar para un bloque actual.

[0106] En algunos ejemplos, el codificador de video 200 y el decodificador de video 300 pueden codificar una lista de ponderaciones en un conjunto de parámetros (como un SPS, un conjunto de parámetros de imagen (PPS), un conjunto de parámetros de video (VPS), un conjunto de parámetros de adaptación (APS), o similares) o encabezados (mosaicos, cortes, unidades de codificación, o similares) correspondientes a los casos descritos en las técnicas anteriores que se basan en verificar modos de bloques vecinos, tamaños de bloques, etc.

[0107] La Figura 4 es un diagrama de bloques que ilustra un codificador de video 200 que puede realizar técnicas de la presente divulgación. La Figura 4 se proporciona para los propósitos de explicación y no se debe considerar limitativo de las técnicas de acuerdo con lo ejemplificado y descrito de manera más amplia en esta divulgación. Para fines de explicación, esta divulgación describe el codificador de video 200 en el contexto de estándares de codificación de video tales como el estándar de codificación de video HEVC y el estándar de codificación de video H.266 en desarrollo. Sin embargo, las técnicas de esta divulgación no se limitan a estos estándares de codificación de video y son aplicables generalmente a la codificación y decodificación de video.

[0108] En el ejemplo de la Figura 4, 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 transformación 206, la unidad de cuantificación 208, la unidad de cuantificación inversa 210, la unidad de procesamiento de transformación inversa 212, la unidad de reconstrucción 214, la unidad de filtro 216, búfer de imagen decodificada (DPB) 218 y unidad de codificación por entropía 220. Cualquiera o toda la memoria de datos de video 230, unidad de selección de modo 202, unidad de generación residual 204, unidad de procesamiento de transformación 206, unidad de cuantificación 208, unidad de cuantificación inversa 210, unidad de procesamiento de transformación inversa 212, unidad de reconstrucción 214, unidad de filtro 216, DPB 218, y la unidad de codificación por entropía 220 puede implementarse en uno o más procesadores o en circuitos de procesamiento. Además, el codificador de video 200 puede incluir procesadores o circuitos de procesamiento adicionales o alternativos para realizar estas y otras funciones.

[0109] La memoria de datos de video 230 puede almacenar los datos de video que van a 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 una memoria de imágenes de referencia que almacena datos de video de referencia para su uso en la predicción de datos de video por el codificador de video 200. La memoria de datos de video 230 y DPB 218 se puede formar mediante cualquiera de una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), lo que incluye DRAM síncrona (SDRAM), RAM magneto resistiva (MRAM), RAM resistiva (RRAM), u otro tipo de dispositivo de memoria. La memoria de datos de video 230 y DPB 218 puede ser proporcionada por el mismo dispositivo de memoria o dispositivos de memoria separados. En diversos ejemplos, la memoria de datos de video 230 puede estar en un chip con otros componentes del codificador de video 200, como se ilustra, o fuera del chip con respecto a esos componentes.

[0110] 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 describa específicamente como tal, o la memoria externa al codificador de video 200, a menos que se describa 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 codificar (por

ejemplo, datos de video para un bloque actual que se va a codificar). La memoria 106 de la Figura 1 también puede proporcionar almacenamiento temporal de salidas de las diversas unidades del codificador de video 200.

[0111] Las diversas unidades de la Figura 4 se ilustran para ayudar a comprender las operaciones realizadas por el 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 los circuitos que proporcionan una funcionalidad particular y están preestablecidos en las operaciones que se pueden realizar. Los circuitos programables se refieren a los 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 circuitos distintos (de función fija o programables) y, en algunos ejemplos, la o más unidades pueden ser circuitos integrados.

[0112] El codificador de video 200 puede incluir unidades lógicas aritméticas (ALU), unidades de función elemental (EFU), circuitos digitales, circuitos analógicos y/o núcleos programables, formados a partir de circuitos programables. En los ejemplos donde las operaciones del codificador de video 200 se realizan mediante el uso de software ejecutado por los circuitos programables, la memoria 106 (Figura 1) puede almacenar el código objeto del software que el codificador de video 200 recibe y ejecuta, u otra memoria dentro del codificador de video 200 (no mostrado) puede almacenar tales instrucciones.

[0113] La memoria de datos de video 230 está configurada 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 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.

[0114] La unidad de selección de modo 202 incluye una 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 la predicción de vídeo de acuerdo con otros modos de 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 lineal (LM), o similar.

[0115] La unidad de selección de modo 202 generalmente coordina múltiples pasadas de codificación para probar combinaciones de parámetros de codificación y valores de distorsión de velocidad resultantes para tales combinaciones. Los parámetros de codificación pueden incluir la partición de CTU en CU, modos de predicción para las CU, tipos de transformación para datos residuales de las CU, parámetros de cuantificación para datos residuales de las CU, etc. La unidad de selección de modo 202 puede finalmente seleccionar la combinación de parámetros de codificación que tienen valores de distorsión de velocidad que son mejores que las otras combinaciones probadas.

[0116] El codificador de video 200 puede dividir una imagen recuperada de la memoria de datos de video 230 en una serie de CTU y encapsular una o más CTU dentro de un segmento. La unidad de selección de modo 202 puede dividir una CTU de la imagen de acuerdo con una estructura de árbol, tal como la estructura QTBT o la estructura de árbol cuaternario de HEVC descrita anteriormente. Como se describió 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".

[0117] En general, la unidad de selección de modo 202 también controla los componentes de esta (por ejemplo, unidad de estimación de movimiento 222, unidad de compensación de movimiento 224 y unidad de intrapredicción 226) para generar un bloque de predicción para un bloque actual (por ejemplo, una CU actual, o en HEVC, la parte 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 que coincidan estrechamente 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 qué tan 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 las diferencias cuadradas

(SSD), la diferencia media absoluta (MAD), las diferencias medias cuadradas (MSD) o similares. La unidad de estimación de movimiento 222 generalmente puede realizar estos cálculos mediante el uso de las 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, lo que indica un bloque de referencia que se asemeja más al bloque actual.

[0118] La unidad de estimación de movimiento 222 puede formar uno o más vectores de movimiento (MV) que definen las posiciones de los bloques de referencia en las imágenes de referencia con respecto a la posición del bloque actual en una imagen actual. La unidad de estimación de movimiento 222 puede entonces proporcionar 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 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 mediante los vectores de movimiento. Por ejemplo, la unidad de compensación de movimiento 224 puede recuperar datos del bloque de referencia mediante el vector de movimiento. Como otro ejemplo, si el vector de movimiento tiene una precisión de muestra fraccionada, la unidad de compensación de movimiento 224 puede interpolar valores para el bloque de predicción de acuerdo con uno o 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, mediante promediado muestra por muestra o promediado ponderado.

[0119] Como otro ejemplo, para la intrapredicción o codificación de intrapredicción, la unidad de intrapredicción 226 puede generar el bloque de predicción a partir de muestras vecinas al bloque actual. Por ejemplo, para los modos direccionales, la unidad de intrapredicción 226 generalmente puede combinar matemáticamente valores de muestras vecinas y completar 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.

[0120] De acuerdo con las técnicas de esta divulgación, la unidad de selección de modo 202 puede realizar una predicción inter-intra de bloques de datos de video (por ejemplo, CU). Es decir, la unidad de selección de modo 202 puede formar un bloque de predicción al hacer que la unidad de estimación de movimiento 222 y la unidad de compensación de movimiento 224 formen un bloque de interpredicción y la unidad de intrapredicción 226 forme un bloque de intrapredicción. La unidad de selección de modo 202 puede formar el bloque de predicción final para un bloque actual al realizar una combinación ponderada de muestras del bloque de interpredicción y el bloque de intrapredicción. Cuando un bloque actual es un bloque de crominancia, la unidad de selección de modo 202 puede determinar ponderaciones para realizar la predicción ponderada de acuerdo con un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intra y/o interpredicción. Es decir, en lugar de determinar las ponderaciones de acuerdo con los bloques vecinos al bloque de crominancia actual, la unidad de selección de modo 202 puede determinar el número de bloques interpredichos vecinos del bloque de luminancia correspondiente al bloque de crominancia actual, luego determinar las ponderaciones en función del número de bloques interpredichos vecinos al bloque de luminancia.

[0121] Como ejemplo, la unidad de selección de modo 202 puede determinar si un bloque vecino por encima y/o un bloque vecino por la izquierda al bloque de luminancia correspondiente están interpredichos. Cuando ninguno de estos bloques vecinos está interpredicho, la unidad de selección de modo 202 puede determinar una ponderación de 3 que se aplicará al bloque de intrapredicción y una ponderación de 1 que se aplicará al bloque de interpredicción. Cuando ambos bloques vecinos están interpredichos, la unidad de selección de modo 202 puede determinar una ponderación de 3 que se aplicará al bloque de interpredicción y una ponderación de 1 que se aplicará al bloque de intrapredicción. Cuando uno de estos bloques vecinos está interpredicho y el otro está intrapredicho, la unidad de selección de modo 202 puede determinar ponderaciones de 2 que se aplicarán tanto al bloque de interpredicción como al bloque de intrapredicción. En algunos ejemplos, la unidad de selección de modo 202 puede contar bloques vecinos predichos mediante el uso de predicción inter-intra y/o copia intrabloque como bloques codificados por interpredicción.

[0122] 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 en bruto, no codificada del bloque actual desde la memoria de datos de video 230 y el bloque de predicción desde 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 muestra por muestra resultantes definen un bloque residual para el bloque actual. En algunos ejemplos, la unidad de generación residual 204 también puede determinar diferencias entre valores de muestra en el bloque residual para generar un bloque residual mediante el uso de modulación por impulsos codificados diferencial residual (RDPCM). En algunos ejemplos, la unidad de generación residual 204 puede formarse mediante el uso de uno o más circuitos sustractores que realizan una resta binaria.

[0123] En los 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 luminancia y las correspondientes unidades de predicción de crominancia. El codificador de video 200 y el decodificador de video 300 puede soportar las PU que tienen diversos tamaños. De acuerdo con lo indicado con anterioridad, el tamaño de una CU se puede referir al tamaño del bloque de codificación de luminancia de la CU y el tamaño de una PU se puede referir al tamaño de una unidad de predicción de luminancia 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 simétricos de PU 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 la partición asimétrica para los tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$, y $nR \times 2N$ de interpredicción.

[0124] En los ejemplos en donde la unidad de selección de modo no divide adicionalmente una CU en PU, cada CU puede estar asociada con un bloque de codificación de luminancia y los correspondientes bloques de codificación de crominancia. De acuerdo con lo anterior, el tamaño de una CU se puede referir al tamaño del bloque de codificación de luminancia de la CU. El codificador de video 200 y el decodificador de video 300 pueden admitir tamaños de CU de $2N \times 2N$, $2N \times N$, o $N \times 2N$.

[0125] Para otras técnicas de codificación de video, como una codificación de modo de copia intrabloque, una codificación de modo afín y una codificación de modo de modelo lineal (LM), como algunos ejemplos, la unidad de selección de modo 202, a través de las unidades respectivas asociadas con las técnicas de codificación, genera un bloque de predicción para el bloque actual que se está codificando. En algunos

ejemplos, como la codificación del modo de paleta, es posible que la unidad de selección de modo 202 no genere un bloque de predicción y, en su lugar, genere elementos de sintaxis que indican la manera de reconstruir el bloque en función de una paleta 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 codificar.

[0126] En algunos ejemplos, la unidad de selección de modo 202 puede seleccionar un modo de predicción inter-intra. En tales ejemplos, la unidad de selección de modo 202 puede ponderar un bloque de interpredicción generado por la unidad de compensación de movimiento 224 y un bloque de intrapredicción generado por la unidad de intrapredicción 226 de acuerdo con cualquiera de las diversas técnicas de esta divulgación. La unidad de selección de modo 202 puede generar un bloque de predicción a partir de los bloques de inter- e intrapredicción ponderados y generar el bloque de predicción generado.

[0127] 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.

[0128] La unidad de procesamiento de transformación 206 aplica una o más transformaciones al bloque residual para generar un bloque de coeficientes de transformación (denominado en la presente "bloque de coeficientes de transformación"). La unidad de procesamiento de transformación 206 puede aplicar varias transformaciones a un bloque residual para formar el bloque de coeficiente de transformación. Por ejemplo, la unidad de procesamiento de transformación 206 puede aplicar una transformación de coseno discreta (DCT), una transformación direccional, una transformación de Karhunen-Loeve (KLT) o una transformación conceptualmente similar a un bloque residual. En algunos ejemplos, la unidad de procesamiento de transformación 206 puede realizar múltiples transformaciones en un bloque residual, por ejemplo, una transformación primaria y una transformación secundaria, tal como una transformación rotacional. En algunos ejemplos, la unidad de procesamiento de transformación 206 no aplica transformaciones a un bloque residual.

[0129] La unidad de cuantificación 208 puede cuantificar los coeficientes de transformación en un bloque de coeficiente de transformación, para producir un bloque de coeficiente de transformación cuantificado. La unidad de cuantificación 208 puede

cuantificar los coeficientes de transformación de un bloque de coeficientes de transformación de acuerdo con un valor de parámetro de cuantificación (QP) asociado con el bloque actual. El codificador de video 200 (por ejemplo, mediante la unidad de selección de modo 202) puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes 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 transformación cuantificados pueden tener una precisión menor que los coeficientes de transformación originales producidos por la unidad de procesamiento de transformación 206.

[0130] La unidad de cuantificación inversa 210 y la unidad de procesamiento de transformación inversa 212 pueden aplicar cuantificación inversa y transformaciones inversas a un bloque de coeficiente de transformación cuantificado, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficiente de transformación. La unidad de reconstrucción 214 puede producir un bloque reconstruido correspondiente al bloque actual (aunque potencialmente con algún grado de distorsión) en función del 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 agregar muestras del bloque residual reconstruido a las muestras correspondientes del bloque de predicción generado por la unidad de selección de modo 202 para producir el bloque reconstruido.

[0131] 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 los artefactos de bloqueo a lo largo de los bordes de las CU. Las operaciones de la unidad de filtro 216 pueden omitirse, en algunos ejemplos.

[0132] El codificador de video 200 almacena bloques reconstruidos en DPB 218. Por ejemplo, en los ejemplos en los que no se necesitan las operaciones de la unidad de filtro 216, la unidad de reconstrucción 214 puede almacenar bloques reconstruidos en DPB 218. En los ejemplos en los que se necesitan 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 interpredecir bloques de imágenes codificadas posteriormente. Además, la unidad de intrapredicción 226 puede usar bloques reconstruidos en DPB 218 de una imagen actual para intrapredicar otros bloques en la imagen actual.

[0133] En general, la unidad de codificación por entropía 220 puede codificar elementos de sintaxis de entropía recibidos de otros componentes funcionales del codificador de video 200. Por ejemplo, la unidad de codificación por entropía 220 puede codificar bloques de coeficientes de transformación cuantificados 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 interpredicción o información de intrepredicción para intrapredicción) de la unidad de selección de modo 202. La unidad de codificación por entropía 220 puede realizar una o más operaciones de codificación por entropía en los elementos de sintaxis, que son otro ejemplo de datos de video, para generar datos codificados por entropía. Por ejemplo, la unidad de codificación por entropía 220 puede llevar a cabo una operación de codificación adaptativa según el contexto de longitud variable (CAVLC), una operación de CABAC, una operación de codificación de longitud de variable a variable (V2V), una operación de codificación aritmética binaria adaptativa según el contexto con base en la sintaxis (SBAC), una operación de codificación por Entropía de Partición de Intervalo de Probabilidad (PIPE), una operación de exponencial de codificación Golomb, u otro tipo de operación de codificación por entropía sobre los datos. En algunos ejemplos, la unidad de codificación por entropía 220 puede funcionar en modo de derivación donde los elementos de sintaxis no están codificados por entropía.

[0134] El codificador de video 200 puede generar 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 por entropía 220 puede generar el flujo de bits.

[0135] Las operaciones descritas anteriormente se describen con respecto a un bloque. Tal descripción debe entenderse como operaciones para un bloque de codificación de luminancia y/o bloques de codificación de crominancia. Como se describió anteriormente, en algunos ejemplos, el bloque de codificación de luminancia y los bloques de codificación de crominancia son componentes de luminancia y crominancia de una CU. En algunos ejemplos, el bloque de codificación de luminancia y los bloques de codificación de crominancia son componentes de luminancia y crominancia de una PU.

[0136] En algunos ejemplos, las operaciones realizadas con respecto a un bloque de codificación de luminancia no necesitan repetirse para los bloques de codificación de crominancia. Como ejemplo, las operaciones para identificar un vector de movimiento

(MV) y una imagen de referencia para un bloque de codificación de luminancia no necesitan repetirse para identificar una MV y una imagen de referencia para los bloques de crominancia. Más bien, el MV para el bloque de codificación de luminancia se puede escalar para determinar el MV para los bloques de crominancia, y la imagen de referencia puede ser la misma. Como otro ejemplo, el proceso de intrapredicción puede ser el mismo para los bloques de codificación de luminancia y los bloques de codificación de crominancia.

[0137] El codificador de video 200 de la Figura 4 representa un ejemplo de un dispositivo para codificar datos de video que comprende uno o más procesadores configurados para formar un bloque de interpredicción para un bloque actual de datos de video; formar un bloque de intrapredicción para el bloque actual de datos de video; aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y codificar (es decir, codificar, en este ejemplo) el bloque actual mediante el uso del bloque de predicción.

[0138] El codificador de video 200 también representa un ejemplo de un dispositivo para codificar datos de video que comprende uno o más procesadores configurados para formar un bloque de interpredicción para un bloque de crominancia actual de datos de video; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el uso del bloque de predicción.

[0139] La Figura 5 es un diagrama de bloques que ilustra un decodificador de video 300 que puede realizar técnicas de la presente divulgación. La Figura 5 se proporciona con fines explicativos y no se limita a las técnicas que se ejemplifican y describen ampliamente en esta divulgación. A efectos de explicación, esta divulgación describe el decodificador de video 300 de acuerdo con las técnicas de JEM, VVC, y HEVC. Sin embargo, las técnicas de esta divulgación pueden realizarse mediante dispositivos de

codificación de video que están configurados para otros estándares de codificación de video.

[0140] En el ejemplo de la Figura 5, el decodificador de video 300 incluye memoria de búfer de imagen codificada (CPB) 320, unidad de decodificación por entropía 302, unidad de procesamiento de predicción 304, unidad de cuantificación inversa 306, unidad de procesamiento de transformación inversa 308, unidad de reconstrucción 310, unidad de filtro 312 y búfer de imagen decodificada (DPB) 314. Cualquiera o toda 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 transformación inversa 308, la unidad de reconstrucción 310, la unidad de filtro 312 y DPB 314 pueden implementarse en uno o más procesadores o en circuitos de procesamiento. Además, el decodificador de video 300 puede incluir procesadores o circuitos de procesamiento adicionales o alternativos para realizar estas y otras funciones.

[0141] La unidad de procesamiento de predicción 304 incluye la unidad de compensación de movimiento 316 y la unidad de intrapredicción 318. La unidad de procesamiento de predicción 304 puede incluir unidades de adición 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 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) o similar. En otros ejemplos, el decodificador de video 300 puede incluir más o menos, o diferentes componentes funcionales.

[0142] En algunos ejemplos, la unidad de procesamiento de predicciones 304 puede seleccionar un modo de predicción inter-intra. En tales ejemplos, la unidad de procesamiento de predicción 304 puede ponderar un bloque de interpredicción generado por la unidad de compensación de movimiento 316 y un bloque de intrapredicción generado por la unidad de intrapredicción 318 de acuerdo con cualquiera de las diversas técnicas de esta divulgación. La unidad de procesamiento de predicción 304 puede generar un bloque de predicción a partir de los bloques de inter- e intrapredicción ponderados y generar el bloque de predicción generado.

[0143] De acuerdo con las técnicas de esta divulgación, la unidad de procesamiento de predicción 304 puede realizar una predicción inter-intra de bloques de datos de video (por ejemplo, CU). Es decir, la unidad de procesamiento de predicción 304 puede formar un bloque de predicción al causar la unidad de compensación de

movimiento 316 forme un bloque de interpredicción y la unidad de intrapredicción 318 forme un bloque de intrapredicción. La unidad de procesamiento de predicción 304 puede formar el bloque de predicción final para un bloque actual al realizar una combinación ponderada de muestras del bloque de interpredicción y el bloque de intrapredicción. Cuando un bloque actual es un bloque de crominancia, la unidad de procesamiento de predicción 304 puede determinar ponderaciones para realizar la predicción ponderada de acuerdo con un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción. Es decir, en lugar de determinar las ponderaciones de acuerdo con los bloques vecinos al bloque de crominancia actual, la unidad de procesamiento de predicción 304 puede determinar el número de bloques intrapredichos vecinos del bloque de luminancia correspondiente al bloque de crominancia actual, luego determinar las ponderaciones en función del número de bloques intrapredichos vecinos al bloque de luminancia.

[0144] Como ejemplo, la unidad de procesamiento de predicción 304 puede determinar si un bloque vecino por encima y/o un bloque vecino por la izquierda al bloque de luminancia correspondiente están interpredichos. Cuando ninguno de estos bloques vecinos está interpredicho, la unidad de procesamiento de predicción 304 puede determinar una ponderación de 3 que se aplicará al bloque de intrapredicción y una ponderación de 1 que se aplicará al bloque de interpredicción. Cuando estos bloques vecinos están interpredichos, la unidad de procesamiento de predicción 304 puede determinar una ponderación de 3 que se aplicará al bloque de interpredicción y una ponderación de 1 que se aplicará al bloque de intrapredicción. Cuando uno de estos bloques vecinos está interpredicho y el otro está intrapredicho, la unidad de procesamiento de predicción 304 puede determinar ponderaciones de 2 que se aplicarán tanto al bloque de interpredicción como al bloque de intrapredicción. En algunos ejemplos, la unidad procesamiento de predicción 304 puede contar bloques vecinos predichos mediante el uso de predicción inter-intra y/o copia intrabloque como bloques codificados por interpredicción.

[0145] La memoria de CPB 320 puede almacenar datos de video, tales como un flujo de bits de video codificado, para 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, del medio legible por ordenador 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 vídeo 300. La DPB 314 generalmente almacena imágenes decodificadas, cuyo decodificador de video 300 puede emitir y/o usar como datos de video de referencia al decodificar datos o imágenes subsiguientes del flujo de bits de video codificado. La memoria de CPB 320 y DPB 314 pueden estar formados por medio de cualquiera de una variedad de dispositivos de memoria, tales como la memoria dinámica de acceso aleatorio (DRAM), que incluyen DRAM síncrona (SDRAM), RAM magneto-resistiva (MRAM), RAM resistiva (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 ejemplos, la memoria de CPB 320 puede estar en un chip con otros componentes del decodificador de video 300, o fuera del chip con respecto a esos componentes.

[0146] Adicional o alternativamente, en algunos ejemplos, el decodificador de video 300 puede recuperar datos de video codificados de la memoria 120 (Figura 1). Es decir, la memoria 120 puede almacenar datos como se analizó anteriormente con la memoria de CPB 320. Asimismo, la memoria 120 puede almacenar instrucciones para ser ejecutadas por el decodificador de video 300, cuando parte o la totalidad de la funcionalidad del decodificador de video 300 se implementa en software para ser ejecutado mediante circuitos de procesamiento del decodificador de video 300.

[0147] Las diversas unidades mostradas en la Figura 5 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. Similar a la Figura 4, los circuitos de función fija se refieren a los circuitos que proporcionan una funcionalidad particular y están preestablecidos en las operaciones que se pueden realizar. Los circuitos programables se refieren a los 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 circuitos

distintos (de función fija o programables) y, en algunos ejemplos, la o más unidades pueden ser circuitos integrados.

[0148] El decodificador de video 300 puede incluir ALU, EFU, circuitos digitales, circuitos analógicos y/o núcleos programables formados a partir de circuitos programables. En los ejemplos en los que las operaciones del decodificador de video 300 se realizan mediante software que se ejecuta en los circuitos programables, la memoria dentro o fuera del chip puede almacenar instrucciones (por ejemplo, código objeto) del software que el decodificador de video 300 recibe y ejecuta.

[0149] 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 transformación inversa 308, la unidad de reconstrucción 310, y la unidad de filtro 312 pueden generar datos de video decodificados con base en los elementos de sintaxis extraídos del flujo de bits.

[0150] 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 actualmente, es decir, decodificado, puede denominarse "bloque actual").

[0151] La unidad de decodificación por entropía 302 puede decodificar por entropía elementos de sintaxis que definen coeficientes de transformación cuantificados de un bloque de coeficientes de transformación cuantificados, así como información de transformación, tal como un parámetro de cuantificación (QP) y/o indicaciones de modo de transformación. La unidad de cuantificación inversa 306 puede usar el QP asociado con el bloque de coeficiente de transformación cuantificado para determinar un grado de cuantificación e, igualmente, un grado de cuantificación inversa para que se aplique 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 forma inversa los coeficientes de transformación cuantificados. La unidad de cuantificación inversa 306 puede formar de ese modo un bloque de coeficientes de transformación que incluye coeficientes de transformación.

[0152] Después de que la unidad de cuantificación inversa 306 forme el bloque de coeficiente de transformación, la unidad de procesamiento de transformación inversa 308 puede aplicar una o más transformaciones inversas al bloque de coeficiente de transformación para generar un bloque residual asociado con el bloque actual. Por ejemplo, la unidad de procesamiento de transformación inversa 308 puede aplicar una

DCT inversa, una transformación inversa de números enteros, una transformación inversa de Karhunen-Loeve (KLT), una transformación inversa de rotación, una transformación direccional inversa, u otra transformación inversa al bloque de coeficientes.

[0153] Además, la unidad de procesamiento de predicción 304 genera un bloque de predicción de acuerdo con los 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 interpretado, 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 de 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 interpretación de una manera que es sustancialmente similar a la descrita con respecto a la unidad de compensación de movimiento 224 (Figura 4).

[0154] 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 4). La unidad de intrapredicción 318 puede recuperar datos de muestras vecinas al bloque actual a partir del DPB 314.

[0155] La unidad de reconstrucción 310 puede reconstruir el bloque actual mediante el uso del bloque de predicción y el bloque residual. Por ejemplo, la unidad de reconstrucción 310 puede agregar muestras del bloque residual a las muestras correspondientes del bloque de predicción para reconstruir el bloque actual.

[0156] 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 bloqueo a lo largo de los bloques reconstruidos. Las operaciones de la unidad de filtro 312 no se realizan necesariamente en todos los ejemplos.

[0157] El decodificador de video 300 puede almacenar los bloques reconstruidos en

DPB 314. Como se analizó anteriormente, DPB 314 puede proporcionar información de referencia, como muestras de una imagen actual para intrapredicción e imágenes previamente decodificadas para compensación de movimiento subsiguiente, a la unidad de procesamiento de predicción 304. Además, el decodificador de video 300 puede emitir imágenes decodificadas a partir del DPB para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 118 de la Figura 1.

[0158] El decodificador de video 300 de la Figura 5 representa un ejemplo de un dispositivo para codificar datos de video que comprende uno o más procesadores configurados para formar un bloque de interpredicción para un bloque actual de datos de video; formar un bloque de intrapredicción para el bloque actual de datos de video; aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y codificar (es decir, decodificar, en este ejemplo) el bloque actual mediante el uso del bloque de predicción.

[0159] El decodificador de video 300 también representa un ejemplo de un dispositivo para codificar datos de video que comprende uno o más procesadores configurados para formar un bloque de interpredicción para un bloque de crominancia actual de datos de video; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el uso del bloque de predicción.

[0160] La Figura 6 es un diagrama de flujo que ilustra un procedimiento de ejemplo para la codificación actual de un bloque de 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 (Figura 1 y 4), debe entenderse que se pueden configurar otros dispositivos para realizar un método similar al de la Figura 6.

[0161] 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 mediante el uso de predicción inter-intra de acuerdo con las técnicas de esta divulgación. 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 y cuantificar los coeficientes del bloque residual (354). A continuación, el codificador de video 200 puede explorar los coeficientes de transformación cuantificados del bloque residual (356). Durante la exploración, o después de la exploración, el codificador de video 200 puede codificar por entropía los coeficientes (358). Por ejemplo, el codificador de video 200 puede codificar los coeficientes mediante el uso de CAVLC o CABAC. El codificador de video 200 puede generar entonces los datos codificados por entropía del bloque (360).

[0162] De esta manera, el método de la Figura 6 representa un ejemplo de un método que incluye la formación de un bloque de interpredicción para un bloque actual de datos de video; formar un bloque de intrapredicción para el bloque actual de datos de video; aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y codificar (es decir, codificar) el bloque actual mediante el uso del bloque de predicción.

[0163] La Figura 7 es un diagrama de flujo que ilustra un procedimiento de ejemplo para la decodificación actual de un bloque 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 5), debe entenderse que se pueden configurar otros dispositivos para realizar un método similar al de la Figura 7.

[0164] 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 un bloque residual correspondiente al bloque actual (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 reproducir los coeficientes del bloque residual (372). El decodificador de video 300 puede predecir el bloque actual (374), por ejemplo, mediante el uso del modo de predicción inter-intra de acuerdo con las técnicas de esta divulgación, como lo indica 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 explorar de forma inversa los coeficientes reproducidos (376), para crear un bloque de coeficientes de transformación cuantificados. El decodificador de video 300 puede entonces cuantificar y transformar inversamente los coeficientes para producir un bloque residual (378). El decodificador de video 300 puede finalmente decodificar el bloque actual al combinar el bloque de predicción y el bloque residual (380).

[0165] De esta manera, el método de la Figura 7 representa un ejemplo de un método que incluye la formación de un bloque de interpredicción para un bloque actual de datos de video; formar un bloque de intrapredicción para el bloque actual de datos de video; aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y codificar (es decir, decodificar) el bloque actual mediante el uso del bloque de predicción.

[0166] La Figura 8 es un diagrama de flujo que ilustra un método de ejemplo de codificación (codificación o decodificación) de datos de video de acuerdo con las técnicas de esta divulgación. Con fines de ejemplo y explicación, el método de la Figura 8 se explica con respecto al decodificador 300 de video de las Figuras 1 y 5, aunque el codificador de video 200 de las Figuras 1 y 4 u otros dispositivos de codificación de video pueden realizar este o un método similar. El método de la Figura 8 puede realizarse mediante el codificador de video 200 cuando se predice un bloque de crominancia actual, por ejemplo, en la etapa 350 del método de la Figura 6, o mediante el decodificador de video 300 cuando se predice un bloque de crominancia actual en, por ejemplo, la etapa 374 de la Figura 7.

[0167] Inicialmente, el decodificador de video 300 forma un bloque de interpredicción para un bloque de crominancia actual (400). El decodificador de video 300 también puede formar un bloque de intrapredicción para el bloque de crominancia actual (402). El decodificador de video 300 puede entonces determinar un bloque de luminancia correspondiente al bloque de crominancia (404), por ejemplo, un bloque de luminancia que se ubica junto con el bloque de crominancia en una matriz de luminancia.

[0168] El decodificador de video 300 puede entonces determinar modos de predicción de bloques vecinos al bloque de luminancia (406). Por ejemplo, el decodificador de video 300 puede determinar los modos de predicción del bloque vecino superior 142 y del bloque vecino izquierdo 144 de la Figura 3A, el bloque vecino superior derecho 148 y el bloque vecino inferior izquierdo 150 de la Figura 3B, o similar para otros ejemplos

de bloques vecinos como se muestra en las Figuras 3A–3F. El decodificador de video 300 también puede determinar un número de bloques vecinos intrapredichos al bloque de luminancia (408). Alternativamente, el decodificador de video 300 puede determinar un número de bloques vecinos interpredichos para el bloque de luminancia.

[0169] El decodificador de video 300 también puede determinar las ponderaciones que se aplicarán al bloque de interpredicción y al bloque de intrapredicción de acuerdo con el número de bloques vecinos intrapredichos al bloque de luminancia (410). Si hay más bloques interpredichos que bloques intrapredichos, el decodificador de video 300 puede determinar que la ponderación que se aplicará al bloque de interpredicción es mayor que la ponderación que se aplicará al bloque de intrapredicción. Por ejemplo, la ponderación que se aplicará al bloque de interpredicción puede ser 3 y la ponderación que se aplicará al bloque de intrapredicción puede ser 1. Si el número de bloques de intrapredicción es mayor que el número de bloques de interpredicción, el decodificador de video 300 puede determinar que la ponderación que se aplicará al bloque de intrapredicción es mayor que la ponderación que se aplicará al bloque de interpredicción. Por ejemplo, la ponderación que se aplicará al bloque de intrapredicción puede ser 3 y la ponderación que se aplicará al bloque de interpredicción puede ser 1. Si el número de bloques de intrapredicción e interpredicción es el mismo, las ponderaciones pueden ser iguales, por ejemplo, 2 para cada uno.

[0170] El decodificador de video 300 puede combinar entonces el bloque de interpredicción y el bloque de intrapredicción mediante el uso de las ponderaciones para formar un bloque de predicción para el bloque de crominancia actual (412). El decodificador de video 300 puede entonces codificar el bloque de crominancia actual mediante el uso del bloque de predicción (414).

[0171] De esta manera, el método de la Figura 8 representa un ejemplo de un método de codificación de datos de video, donde el método comprende formar un bloque de interpredicción para un bloque de crominancia actual de datos de video; formar un bloque de intrapredicción para el bloque de crominancia actual de los datos de video; determinar un número de bloques vecinos a un bloque de luminancia correspondiente al bloque de crominancia actual que están codificados por intrapredicción; determinar una primera ponderación y una segunda ponderación de acuerdo con el número de bloques vecinos que están codificados por intrapredicción; aplicar la primera ponderación al bloque de interpredicción y la segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo

bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque de crominancia actual; y codificar el bloque de crominancia actual mediante el bloque de predicción.

[0172] Ciertas técnicas de esta divulgación se resumen en los siguientes ejemplos:

[0173] Ejemplo 1: Un método para codificar datos de video, donde el método comprende: formar un bloque de interpredicción para un bloque actual de datos de video; formar un bloque de intrapredicción para el bloque actual de datos de video; aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y codificar el bloque actual mediante el uso del bloque de predicción.

[0174] Ejemplo 2: El método de acuerdo con el ejemplo 1, en donde codificar el bloque actual comprende decodificar el bloque actual, que comprende: decodificar un bloque residual para el bloque actual; y combinar muestras del bloque residual con muestras del bloque de predicción.

[0175] Ejemplo 3: El método de acuerdo con cualquiera de los ejemplos 1 y 2, en donde codificar el bloque actual comprende codificar el bloque actual, que comprende: restar muestras del bloque de predicción de muestras del bloque actual para formar un bloque residual; y codificar el bloque residual.

[0176] Ejemplo 4: El método de acuerdo con cualquiera de los ejemplos 1–3, que comprende además determinar la primera ponderación y la segunda ponderación de acuerdo con un número de bloques vecinos intracodificados al bloque actual.

[0177] Ejemplo 5: El método de acuerdo con el ejemplo 4, en donde los bloques vecinos incluyen un bloque vecino superior.

[0178] Ejemplo 6: El método de acuerdo con cualquiera de los ejemplos 4 y 5, en donde los bloques vecinos incluyen un bloque vecino izquierdo.

[0179] Ejemplo 7: El método de acuerdo con cualquiera de los ejemplos 4–6, en donde los bloques vecinos incluyen un bloque vecino superior derecho.

[0180] Ejemplo 8: El método de acuerdo con cualquiera de los ejemplos 4–7, en donde los bloques vecinos incluyen un bloque vecino inferior izquierdo.

[0181] Ejemplo 9: El método de acuerdo con cualquiera de los ejemplos 4–8, en donde los bloques vecinos incluyen un bloque vecino superior izquierdo.

[0182] Ejemplo 10: El método de acuerdo con cualquiera de los ejemplos 4–9, que comprende además determinar los bloques vecinos de acuerdo con la altura del

bloque actual y el ancho del bloque actual.

[0183] Ejemplo 11: El método de acuerdo con el ejemplo 10, en donde cuando la altura y el ancho son iguales, los bloques vecinos incluyen un bloque vecino superior derecho y un bloque vecino inferior izquierdo.

[0184] Ejemplo 12: El método de acuerdo con el ejemplo 11, en donde los bloques vecinos incluyen además un bloque vecino superior y un bloque vecino izquierdo.

[0185] Ejemplo 13: El método de acuerdo con el ejemplo 12, en donde los bloques vecinos incluyen además un bloque vecino superior izquierdo.

[0186] Ejemplo 14: El método de acuerdo con cualquiera de los ejemplos 10–13, en donde cuando la altura y el ancho son diferentes, los bloques vecinos incluyen un bloque vecino superior derecho y un bloque vecino inferior izquierdo.

[0187] Ejemplo 15: El método de acuerdo con el ejemplo 10, en donde cuando la altura y el ancho son iguales, los bloques vecinos son simétricos a lo largo de las dimensiones de ancho y altura del bloque actual, y cuando la altura y el ancho no son iguales, los bloques vecinos son asimétricos a lo largo de las dimensiones del ancho y la altura del bloque actual.

[0188] Ejemplo 16: El método de acuerdo con el ejemplo 15, en donde cuando el ancho es mayor que la altura, los bloques vecinos incluyen un bloque vecino superior derecho y un bloque vecino izquierdo, y cuando la altura es mayor que el ancho, los bloques vecinos incluyen un bloque vecino superior y un bloque vecino inferior izquierdo.

[0189] Ejemplo 17: El método de acuerdo con cualquiera de los ejemplos 4–16, que comprende además determinar que uno de los bloques vecinos que está codificado por copia intrabloque o codificado por referencia de imagen actual es un bloque intercodificado.

[0190] Ejemplo 18: El método de acuerdo con cualquiera de los ejemplos 4–16, que comprende además determinar que uno de los bloques vecinos que está codificado por copia intrabloque o codificado por referencia de imagen actual es un bloque intracodificado.

[0191] Ejemplo 19: El método de acuerdo con cualquiera de los ejemplos 4–18, que comprende además determinar que uno de los bloques vecinos que está codificado por inter-intra es un bloque intercodificado.

[0192] Ejemplo 20: El método de acuerdo con cualquiera de los ejemplos 4–18, que comprende además determinar que uno de los bloques vecinos que está codificado por inter-intra es un bloque intracodificado.

[0193] Ejemplo 21: El método de acuerdo con cualquiera de los ejemplos 4–20, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la segunda ponderación para que sea mayor que la primera ponderación cuando todos los bloques vecinos están intracodificados.

[0194] Ejemplo 22: El método de acuerdo con cualquiera de los ejemplos 4–21, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación para que sea mayor que la segunda ponderación cuando al menos uno de los bloques vecinos no está intracodificado.

[0195] Ejemplo 23: El método de acuerdo con cualquiera de los ejemplos 4–22, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación para que sea igual que la segunda ponderación cuando solo uno de los bloques vecinos está intracodificado.

[0196] Ejemplo 24: El método de acuerdo con cualquiera de los ejemplos 4–23, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 1 y la segunda ponderación como 3 cuando un bloque vecino superior izquierdo, un bloque vecino superior derecho, y un bloque vecino inferior izquierdo al bloque actual están intracodificados.

[0197] Ejemplo 25: El método de acuerdo con cualquiera de los ejemplos 4–23, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 3 y la segunda ponderación como 5 cuando un bloque vecino superior izquierdo, un bloque vecino superior derecho, y un bloque vecino inferior izquierdo al bloque actual están intracodificados.

[0198] Ejemplo 26: El método de acuerdo con cualquiera de los ejemplos 4–25, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 2 y la segunda ponderación como 2 cuando al menos uno de un bloque vecino superior izquierdo, un bloque vecino superior derecho, o un bloque vecino inferior izquierdo al bloque actual está intracodificado.

[0199] Ejemplo 27: El método de acuerdo con cualquiera de los ejemplos 4–25, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 4 y la segunda ponderación como 4 cuando al menos uno de un bloque vecino superior izquierdo, un bloque vecino superior derecho, o un bloque vecino inferior izquierdo al bloque actual está intracodificado.

[0200] Ejemplo 28: El método de acuerdo con cualquiera de los ejemplos 4–27, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 3 y la segunda ponderación como 1 cuando

ninguno de un bloque vecino superior izquierdo, un bloque vecino superior derecho, y un bloque vecino inferior izquierdo al bloque actual está intracodificado.

[0201] Ejemplo 29: El método de acuerdo con cualquiera de los ejemplos 4–27, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación como 5 y la segunda ponderación como 3 cuando ninguno de un bloque vecino superior izquierdo, un bloque vecino superior derecho, y un bloque vecino inferior izquierdo al bloque actual está intracodificado.

[0202] Ejemplo 30: El método de acuerdo con cualquiera de los ejemplos 4–29, en donde determinar la primera ponderación y la segunda ponderación comprende copiar la primera ponderación y la segunda ponderación de uno de los bloques vecinos que está intracodificado por múltiples hipótesis.

[0203] Ejemplo 31: El método de acuerdo con cualquiera de los ejemplos 4–30, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la segunda ponderación para que sea mayor que la primera velocidad cuando solo uno de los bloques vecinos está intracodificado.

[0204] Ejemplo 32: El método de acuerdo con el ejemplo 31, en donde la segunda ponderación es 3 y la primera ponderación es 1.

[0205] Ejemplo 33: El método de acuerdo con el ejemplo 31, en donde la segunda ponderación es 5 y la primera ponderación es 3.

[0206] Ejemplo 34: El método de acuerdo con cualquiera de los ejemplos 4–30, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la primera ponderación para que sea mayor que la segunda velocidad cuando solo uno de los bloques vecinos está intracodificado.

[0207] Ejemplo 35: El método de acuerdo con el ejemplo 34, en donde la primera ponderación es 3 y la segunda ponderación es 1.

[0208] Ejemplo 36: El método de acuerdo con el ejemplo 34, en donde la primera ponderación es 5 y la segunda ponderación es 3.

[0209] Ejemplo 37: El método de acuerdo con cualquiera de los ejemplos 4–30, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la segunda ponderación para que sea mayor que la primera velocidad cuando ninguno de los bloques vecinos está intracodificado.

[0210] Ejemplo 38: El método de acuerdo con el ejemplo 37, en donde la segunda ponderación es 3 y la primera ponderación es 1.

[0211] Ejemplo 39: El método de acuerdo con el ejemplo 37, en donde la segunda ponderación es 5 y la primera ponderación es 3.

[0212] Ejemplo 40: El método de acuerdo con cualquiera de los ejemplos 4–30, en donde determinar la primera ponderación y la segunda ponderación comprende seleccionar la segunda ponderación para que sea mayor que la primera velocidad cuando al menos uno de los bloques vecinos está predicho mediante modo intra DC o modo planar.

[0213] Ejemplo 41: El método de acuerdo con el ejemplo 40, en donde la segunda ponderación es 3 y la primera ponderación es 1.

[0214] Ejemplo 42: El método de acuerdo con el ejemplo 40, en donde la segunda ponderación es 5 y la primera ponderación es 3.

[0215] Ejemplo 43: El método de acuerdo con cualquiera de los ejemplos 1–42, que comprende además determinar la primera ponderación y la segunda ponderación de acuerdo con un bloque de luminancia de uno de los bloques vecinos cuando uno de los bloques vecinos tiene un árbol de codificación luminancia-crominancia único o árbol de codificación luminancia-crominancia dual habilitado.

[0216] Ejemplo 44: El método de acuerdo con cualquiera de los ejemplos 1–43, que comprende además determinar la primera ponderación y la segunda ponderación de acuerdo con un bloque de crominancia de uno de los bloques vecinos cuando uno de los bloques vecinos tiene un árbol de codificación luminancia-crominancia dual habilitado.

[0217] Ejemplo 45: El método de acuerdo con cualquiera de los ejemplos 1–44, que comprende además determinar la primera ponderación y la segunda ponderación de acuerdo con una serie de interpretaciones de un bloque indexado de fusión.

[0218] Ejemplo 46: El método de acuerdo con el ejemplo 45, en donde determinar la primera ponderación y la segunda ponderación comprende establecer la primera ponderación mayor que la segunda ponderación cuando el bloque indexado de fusión es bipredicción.

[0219] Ejemplo 47: El método de acuerdo con el ejemplo 46, en donde la primera ponderación es 3 y la segunda ponderación es 1.

[0220] Ejemplo 48: El método de acuerdo con el ejemplo 46, en donde la primera ponderación es 5 y la segunda ponderación es 3.

[0221] Ejemplo 49: El método de acuerdo con cualquiera de los ejemplos 46–48, en donde determinar la primera ponderación y la segunda ponderación comprende determinar la primera ponderación y la segunda ponderación de acuerdo con uno de los ejemplos 4–42 cuando el bloque indexado de fusión es unipredicción.

[0222] Ejemplo 50: El método de acuerdo con el ejemplo 45, en donde determinar la

primera ponderación y la segunda ponderación comprende establecer la primera ponderación mayor que la segunda ponderación cuando el bloque indexado de fusión es unipredicción.

[0223] Ejemplo 51: El método de acuerdo con el ejemplo 50, en donde la primera ponderación es 3 y la segunda ponderación es 1.

[0224] Ejemplo 52: El método de acuerdo con el ejemplo 50, en donde la primera ponderación es 5 y la segunda ponderación es 3.

[0225] Ejemplo 53: El método de acuerdo con cualquiera de los ejemplos 1–52, que comprende además determinar la primera ponderación y la segunda ponderación de acuerdo con un tamaño del bloque actual.

[0226] Ejemplo 54: El método de acuerdo con el ejemplo 53, en donde el tamaño del bloque actual comprende un mínimo del ancho del bloque actual por la altura del bloque actual.

[0227] Ejemplo 55: El método de acuerdo con el ejemplo 53, en donde el tamaño del bloque actual comprende un máximo del ancho del bloque actual por la altura del bloque actual.

[0228] Ejemplo 56: El método de acuerdo con el ejemplo 53, en donde el tamaño del bloque actual comprende un producto del ancho del bloque actual por la altura del bloque actual.

[0229] Ejemplo 57: El método de acuerdo con el ejemplo 53, en donde el tamaño del bloque actual comprende una suma del ancho del bloque actual más la altura del bloque actual.

[0230] Ejemplo 58: El método de acuerdo con cualquiera de los ejemplos 53–57, en donde determinar la primera ponderación y la segunda ponderación comprende establecer la primera ponderación mayor que la segunda ponderación cuando el tamaño del bloque es mayor que un umbral predefinido.

[0231] Ejemplo 59: El método de acuerdo con el ejemplo 58, en donde la primera ponderación es 3 y la segunda ponderación es 1.

[0232] Ejemplo 60: El método de acuerdo con el ejemplo 58, en donde la primera ponderación es 5 y la segunda ponderación es 3.

[0233] Ejemplo 61: El método de acuerdo con cualquiera de los ejemplos 53–57, en donde determinar la primera ponderación y la segunda ponderación comprende establecer la segunda ponderación mayor que la primera ponderación cuando el tamaño del bloque es mayor que un umbral predefinido.

[0234] Ejemplo 62: El método de acuerdo con el ejemplo 61, en donde la segunda

ponderación es 3 y la primera ponderación es 1.

[0235] Ejemplo 63: El método de acuerdo con el ejemplo 61, en donde la segunda ponderación es 5 y la primera ponderación es 3.

[0236] Ejemplo 64: El método de acuerdo con cualquiera de los ejemplos 53–57, en donde determinar la primera ponderación y la segunda ponderación comprende establecer la primera ponderación mayor que la segunda ponderación cuando el tamaño del bloque es menor que un umbral predefinido.

[0237] Ejemplo 65: El método de acuerdo con el ejemplo 64, en donde la primera ponderación es 3 y la segunda ponderación es 1.

[0238] Ejemplo 66: El método de acuerdo con el ejemplo 64, en donde la primera ponderación es 5 y la segunda ponderación es 3.

[0239] Ejemplo 67: El método de acuerdo con cualquiera de los ejemplos 53–57, en donde determinar la primera ponderación y la segunda ponderación comprende establecer la segunda ponderación mayor que la primera ponderación cuando el tamaño del bloque es menor que un umbral predefinido.

[0240] Ejemplo 68: El método de acuerdo con el ejemplo 67, en donde la segunda ponderación es 3 y la primera ponderación es 1.

[0241] Ejemplo 69: El método de acuerdo con el ejemplo 67, en donde la segunda ponderación es 5 y la primera ponderación es 3.

[0242] Ejemplo 70: El método de acuerdo con cualquiera de los ejemplos 1–69, que comprende además codificar datos de un flujo de bits que representa la primera ponderación y la segunda ponderación.

[0243] Ejemplo 71: El método de acuerdo con el ejemplo 70, en donde los datos del flujo de bits comprenden al menos uno de un conjunto de parámetros de secuencia, un conjunto de parámetros de imagen, un conjunto de parámetros de adaptación, un conjunto de parámetros de video, un encabezado de imagen, un encabezado de segmento, un encabezado de mosaico, o un encabezado de bloque.

[0244] Ejemplo 72: Un dispositivo para codificar datos de video, en donde el dispositivo comprende uno o más medios para realizar el método de acuerdo con cualquiera de los ejemplos 1-71.

[0245] Ejemplo 73: El dispositivo de acuerdo con el ejemplo 72, en donde uno o más medios comprenden uno o más procesadores implementados en circuitos.

[0246] Ejemplo 74: El dispositivo de acuerdo con el ejemplo 72, que comprende además una pantalla configurada para mostrar datos de video decodificados.

[0247] Ejemplo 75: El dispositivo de acuerdo con el ejemplo 72, en donde el

dispositivo comprende uno o más de una cámara, una computadora, un dispositivo móvil, un dispositivo receptor de transmisión o un decodificador.

[0248] Ejemplo 76: El dispositivo de acuerdo con el ejemplo 72, que comprende además una memoria configurada para almacenar datos de video.

[0249] Ejemplo 77: Un dispositivo para codificar datos de video, en donde el dispositivo comprende: medios para formar un bloque de interpredicción para un bloque actual de datos de video; medios para formar un bloque de intrapredicción para el bloque actual de datos de video; medios para aplicar una primera ponderación al bloque de interpredicción y una segunda ponderación al bloque de intrapredicción; medios para combinar el primer bloque de interpredicción ponderado y el segundo bloque de intrapredicción ponderado para formar un bloque de predicción para el bloque actual; y medios para codificar el bloque actual mediante el uso del bloque de predicción.

[0250] Ejemplo 78: Un medio de almacenamiento legible por computadora que tiene almacenadas instrucciones que, cuando se ejecutan, hacen que un procesador realice el método de acuerdo con cualquiera de los ejemplos 1-71.

[0251] Cabe aclarar que según el ejemplo, algunos actos o eventos de cualquiera de las técnicas descritas en la presente se pueden realizar en una secuencia diferente, se pueden agregar, fusionar o excluir completamente (por ejemplo, no todos los actos o eventos descritos son necesarios para la práctica de las técnicas). Asimismo, en algunos ejemplos, los actos o eventos se pueden realizar de manera concurrente, por ejemplo, a través de procesamiento multihilo, procesamiento interrumpido o procesadores múltiples, en lugar de secuencialmente.

[0252] 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 como una o más instrucciones o código en un medio legible por computadora y pueden ejecutarse mediante una unidad de procesamiento de hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como un medio de almacenamiento de datos, o medios de comunicación incluyendo cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador generalmente pueden corresponder a (1) medios de almacenamiento legibles por ordenador tangibles no transitorios o (2) un medio de comunicación tal como una onda de señal o

portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se puede acceder mediante una o más computadoras o uno o más procesadores para obtener instrucciones, códigos y/o estructuras de datos para la implementación de las técnicas descritas en la presente descripción. Un producto de programa de ordenador puede incluir un medio legible por ordenador.

[0253] A modo ejemplificativo, y no limitativo, tal medio de almacenamiento legible por computadora puede comprender RAM, ROM, EEPROM, CD-ROM u otros dispositivos de almacenamiento en disco óptico, de almacenamiento en disco magnético u otro almacenamiento magnético, o cualquier otro medio que pueda utilizarse para almacenar el código de programación deseado en forma de instrucciones o estructuras de datos y al que puede accederse por medio de una computadora. Además, cualquier conexión se denomina de manera adecuada medio legible por computadora. Por ejemplo, si las instrucciones se transmiten desde un sitio web, servidor u otra fuente remota mediante un cable coaxial, cable de fibra óptica, par trenzado, línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarroja, radio y microondas, entonces el cable coaxial, cable de fibra óptica, par trenzado, DSL o tecnologías inalámbricas tales como infrarroja, radio y microondas están incluidas en la definición de medio. Sin embargo, cabe aclarar que los medios de almacenamiento legibles por computadora no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en su lugar, son dirigidos a medios de almacenamiento tangibles no transitorios. Como se usa en la presente, disco incluye disco compacto (CD), disco láser, disco óptico, disco versátil digital (DVD), disquete y disco Blu-ray, en donde los discos reproducen datos magnéticamente y datos ópticamente con láseres. También deberían incluirse combinaciones de lo anterior dentro del alcance de medios legibles por computadora.

[0254] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señal digital (DSP), procesadores de propósito general, circuitos integrados para aplicaciones específicas (ASIC), matrices de compuerta programable en campo (FPGA) u otro circuito lógico integrado o separado equivalente. En consecuencia, los términos "procesador" y "circuito de procesamiento", como se utiliza en la presente puede referirse a cualquiera de las estructuras anteriores o cualquier otra estructura adecuada para la aplicación de las técnicas descritas en la presente. Además, en algunos aspectos, la funcionalidad descrita en la presente puede proporcionarse dentro de los módulos de hardware y/o software dedicados configurados para codificar y decodificar, o incorporarse en un códec

combinado. Además, las técnicas podrían implementarse totalmente en uno o más circuitos o elementos lógicos.

[0255] Las técnicas de la presente descripción se pueden implementar en una amplia variedad de dispositivos o aparatos, que incluyen un microteléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). En la presente descripción se describen distintos componentes, módulos o unidades para enfatizar los aspectos funcionales de los dispositivos configurados para realizar las técnicas descritas, pero no es necesario que sean realizadas por diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, las diversas unidades pueden ser combinadas en una unidad de hardware de códec o proporcionadas por una colección de unidades de hardware interoperativas, lo que incluye uno o más procesadores como se describe anteriormente, en conjunción con el software y/o firmware adecuado.

[0256] Se han descrito distintos ejemplos. Estos y otros ejemplos se encuentran dentro del alcance de las siguientes reivindicaciones.