

PREDICCIÓN TEMPORAL DE FILTRADO EN BUCLE ADAPTATIVO MODIFICADA PARA SOPORTE DE ESCALABILIDAD

[0001] La presente solicitud reivindica el beneficio de la solicitud provisional estadounidense N.º 62/442.322, presentada el 04 de enero de 2017, y la solicitud provisional estadounidense N.º 62/445.174, presentada el 11 de enero de 2017, cuyos contenidos completos se incorporan en la presente por referencia.

CAMPO TÉCNICO

[0002] La presente divulgación se refiere a la codificación y decodificación de video.

ANTECEDENTES

[0003] Las capacidades de video digital se pueden incorporar a un amplio rango de dispositivos, que incluyen televisores digitales, sistemas digitales de transmisión directa, sistemas de transmisión inalámbricos, asistentes digitales personales (PDA), computadoras portátiles o de escritorio, *tablets*, *e-readers*, cámaras digitales, dispositivos de registro digitales, transmisores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, radioteléfonos celulares o satelitales, denominados "*smartphones*", dispositivos de videoteleconferencia, dispositivos de *streaming* de videos, y similares. Los dispositivos de video digital implementan técnicas de compresión 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, el estándar de 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 compresión de video.

[0004] Las técnicas de compresión de video pueden llevar a cabo 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 en bloques, un segmento (*slice*) de video (por ejemplo, un cuadro de video o una porción de un cuadro de video) se puede particionar en bloques de video, tales como bloques de árbol (*treeblocks*) de codificación y bloques de codificación. La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se desea codificar. Los datos residuales representan las diferencias de píxeles entre el bloque original que se desea codificar y el bloque predictivo. Para una mayor compresión, los

datos residuales se pueden transformar del dominio de píxeles a un dominio de la transformada, lo que da como resultado coeficientes de la transformada residuales, que luego se pueden cuantificar.

SÍNTESIS

[0005] En general, la presente divulgación describe técnicas relacionadas al filtrado en bucle adaptativo (ALF), en especial para la predicción de filtros de ALF de cuadros, segmentos o mosaicos codificados de forma previa. Las técnicas se pueden usar en el contexto de códecs de video avanzado, tales como extensiones de HEVC o la próxima generación de estándares de codificación de video.

[0006] En un ejemplo, la presente divulgación describe un método para decodificar datos de video, en donde el método comprende: recibir una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenar, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; y aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual.

[0007] En otro ejemplo, la presente divulgación describe un método para codificar datos de video, en donde el método comprende: generar una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenar, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que

corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinar, en función de un conjunto seleccionado de parámetros de ALF en una de las matrices correspondientes a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual; y, después de aplicar el filtrado en bucle adaptativo a la región actual, usar la región actual para la predicción de una imagen posterior de los datos de video.

[0008] En otro ejemplo, la presente divulgación describe un dispositivo para decodificar datos de video, en donde el dispositivo comprende uno o más medios de almacenamiento configurados para almacenar los datos de video; y uno o más procesadores configurados para: recibir una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenar, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; y aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo para la región actual.

[0009] En otro ejemplo, la presente divulgación describe un dispositivo para codificar datos de video, en donde el dispositivo comprende uno o más medios de almacenamiento configurados para almacenar los datos de video; y uno o más procesadores configurados para: generar una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenar, en la matriz respectiva, conjuntos de parámetros de filtrado en

bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinar, en función de un conjunto seleccionado de parámetros de ALF en una de las matrices correspondientes a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual; y, después de aplicar el filtrado en bucle adaptativo a la región actual, usar la región actual para la predicción de una imagen posterior de los datos de video.

[0010] En otro ejemplo, la presente divulgación describe un dispositivo para decodificar datos de video, en donde el dispositivo comprende: medios para recibir una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; medios para reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, medios para almacenar, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; medios para determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; y medios para aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual.

[0011] En otro ejemplo, la presente divulgación describe un dispositivo para codificar datos de video, en donde el dispositivo comprende: medios para generar una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; medios para reconstruir la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, medios para almacenar, en

la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; medios para determinar, en función de un conjunto seleccionado de parámetros de ALF en una de las matrices correspondientes a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; medios para aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual; y, medios para usar, después de aplicar el filtrado en bucle adaptativo a la región actual, la región actual para la predicción de una imagen posterior de los datos de video.

[0012] En otro ejemplo, la presente divulgación describe un medio de almacenamiento de datos legible por computadora que almacena instrucciones que, cuando se ejecutan, provocan que uno o más procesadores: reciban una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruyan la imagen actual; para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenen, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinen, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; y apliquen, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo para la región actual.

[0013] En otro ejemplo, la presente divulgación describe un medio de almacenamiento legible por computadora que almacena instrucciones que, cuando se ejecutan, provocan que uno o más procesadores: generen una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video, en donde una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual; reconstruyan la imagen actual; para cada

matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, almacenen, en la matriz respectiva, conjuntos de parámetros de filtrado en bucle adaptativo (ALF) que se utilizan en la aplicación de filtros de ALF a las muestras de regiones de imágenes de los datos de video que se decodifican antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o en una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva; determinen, en función de un conjunto seleccionado de parámetros de ALF en una de las matrices correspondientes a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual; apliquen, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo para la región actual; y, después de aplicar el filtrado en bucle adaptativo a la región actual, usen la región actual para la predicción de una imagen posterior de los datos de video.

[0014] Los detalles de uno o más aspectos de la divulgación se indican en los dibujos adjuntos y la descripción a continuación. Otras características, objetos y ventajas de las técnicas descritas en la presente divulgación serán evidentes de la descripción, dibujos y reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0015] La Figura 1 es un diagrama de bloques que ilustra un ejemplo de sistema de codificación y decodificación de video que puede usar una o más técnicas descritas en la presente divulgación.

[0016] La Figura 2 ilustra tres soportes de filtros de Filtrado en Bucle Adaptativo (ALF) de ejemplo diferentes.

[0017] La Figura 3 ilustra un ejemplo de configuración de acceso aleatorio con Grupo de Imágenes (GOP) igual a 16.

[0018] La Figura 4A ilustra una matriz para almacenar parámetros de filtro.

[0019] La Figura 4B ilustra un estado diferente de la matriz para almacenar parámetros de filtro.

[0020] La Figura 5 ilustra una pluralidad de matrices que corresponden a diferentes capas temporales, de acuerdo con una primera técnica de la presente divulgación.

[0021] La Figura 6 ilustra una matriz para almacenar parámetros de ALF y valores del índice de capa temporal asociados, de acuerdo con una segunda técnica de la presente divulgación.

[0022] La Figura 7 es un diagrama de bloques que ilustra un codificador de video de ejemplo que puede implementar una o más técnicas descritas en la presente

divulgación.

[0023] La Figura 8 es un diagrama de bloques que ilustra un decodificador de video de ejemplo que puede implementar una o más técnicas descritas en la presente divulgación.

[0024] La Figura 9 es un diagrama de flujo que ilustra una operación de ejemplo de un codificador de video, de acuerdo con una primera técnica de la presente divulgación.

[0025] La Figura 10 es un diagrama de flujo que ilustra una operación de ejemplo de un decodificador de video, de acuerdo con la primera técnica de la presente divulgación.

[0026] La Figura 11 es un diagrama de flujo que ilustra una operación de ejemplo de un codificador de video, de acuerdo con una segunda técnica de la presente divulgación.

[0027] La Figura 12 es un diagrama de flujo que ilustra una operación de ejemplo de un decodificador de video, de acuerdo con la segunda técnica de la presente divulgación.

DESCRIPCIÓN DETALLADA

[0028] El Filtrado en Bucle Adaptativo (ALF) es un proceso que aplica uno o más filtros adaptativos (es decir, filtros de ALF) como parte de un bucle de codificación para mejorar la calidad de los datos de video decodificados. Un filtro de ALF se asocia a un conjunto de coeficientes. Un codificador de video (es decir, un codificador de video o un decodificador de video) puede aplicar filtros de ALF con diferentes coeficientes a diferentes bloques de la misma imagen, en función de las características de los bloques. Para reducir la sobrecarga asociada con la señalización de los coeficientes asociados a los filtros de ALF, un codificador de video puede almacenar, en una matriz, conjuntos de parámetros de ALF para los filtros de ALF usados en imágenes, mosaicos o segmentos codificados de forma previa. Un conjunto de parámetros de ALF puede incluir múltiples coeficientes asociados a uno o más filtros de ALF. Por ejemplo, un conjunto de parámetros de ALF puede indicar coeficientes asociados a múltiples filtros. El codificador de video reemplaza conjuntos de parámetros de ALF en la matriz de acuerdo con el esquema Primero en Entrar, Primero en Salir (FIFO).

[0029] Las diferentes imágenes en una secuencia de video pueden pertenecer a diferentes capas temporales. Las diferentes capas temporales se asocian a diferentes identificadores temporales. Una imagen en una capa temporal determinada se puede decodificar con referencia a otras imágenes que tienen el identificador temporal de la capa temporal determinada e imágenes que tienen identificadores temporales con

valores inferiores a los del identificador temporal de la capa temporal determinada.

[0030] Puesto que un codificador de video almacena datos del filtro (por ejemplo, conjuntos de parámetros de ALF) en la matriz de acuerdo con un esquema de FIFO, la matriz puede contener datos del filtro de una imagen que tiene un identificador temporal superior a un identificador temporal de una imagen que se está codificando en la actualidad. Esto podría provocar, de forma potencial, errores en el proceso de filtrado ya que podría resultar en que la imagen actual sea dependiente de una imagen en una capa temporal con un identificador temporal superior que la capa temporal de la imagen actual si la imagen con el identificador temporal superior se pierde o no se necesita decodificar.

[0031] La presente divulgación describe técnicas que podrían abordar este inconveniente. En un ejemplo, un codificador de video puede almacenar, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de uno o más filtros de ALF a muestras de regiones de imágenes de los datos de video codificados con anterioridad a la imagen actual. Cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal distinta respectiva. Además, el codificador de video puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a una capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. La presente divulgación puede usar el término “región” para referirse a un segmento u otro tipo de área de una imagen actual para llevar a cabo el ALF. El codificador de video puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, un filtro de ALF a la región actual.

[0032] En algunos ejemplos, un codificador de video almacena, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de uno o más filtros de ALF a muestras de imágenes de los datos de video decodificados con anterioridad a la imagen actual. De forma adicional, en el presente ejemplo, el codificador de video almacena, en la matriz, índices de capa temporal asociados a los conjuntos de parámetros de ALF. Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. En este ejemplo, el codificador de video puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Además, en este ejemplo, el codificador de video puede aplicar, en función del

conjunto de parámetros de ALF aplicable para la región actual, el filtrado en bucle adaptativo a la región actual.

[0033] En cualquiera de estos ejemplos, la asociación de parámetros de ALF con capas temporales puede ayudar a evitar el problema de que una imagen actual sea dependiente en forma potencial de la decodificación de una imagen en una capa temporal superior.

[0034] La Figura 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de video 10 de ejemplo que puede usar técnicas de la presente divulgación. Como se muestra en la Figura 1, el sistema 10 incluye un dispositivo fuente 12 que proporciona datos de video codificados que se desean decodificar posteriormente mediante un dispositivo de destino 14. En particular, el dispositivo fuente 12 proporciona los datos de video codificados al dispositivo de destino 14 mediante un medio legible por computadora 16. El dispositivo fuente 12 y el dispositivo de destino 14 pueden comprender cualquier dispositivo del amplio rango de dispositivos, que incluyen computadoras de escritorio, *notebooks* (es decir, computadoras portátiles), *tablets*, decodificadores de televisión, microteléfonos, tales como los denominados teléfonos “inteligentes”, *tablets*, televisiones, cámaras, dispositivos de visualización, transmisores de medios digitales, consolas de videojuegos, dispositivos de *streaming* de videos o similares. En algunos casos, el dispositivo fuente 12 y el dispositivo de destino 14 están equipados para la comunicación inalámbrica. Por lo tanto, el dispositivo fuente 12 y el dispositivo de destino 14 pueden ser dispositivos de comunicación inalámbricos. Las técnicas descritas en la presente divulgación se pueden aplicar a aplicaciones inalámbricas y/o alámbricas. El dispositivo fuente 12 es un dispositivo de codificación de video de ejemplo (es decir, un dispositivo para codificar datos de video). El dispositivo de destino 14 es un dispositivo de decodificación de video de ejemplo (es decir, un dispositivo para decodificar datos de video).

[0035] El sistema 10 ilustrado de la Figura 1 es solo un ejemplo. Las técnicas para procesar datos de video se pueden llevar a cabo mediante cualquier dispositivo de codificación y/o decodificación de video digital. En algunos ejemplos, las técnicas se pueden llevar a cabo mediante un codificador de video, denominado generalmente “CÓDEC.” El dispositivo fuente 12 y el dispositivo de destino 14 son ejemplos de los dispositivos de codificación, en donde el dispositivo fuente 12 genera datos de video codificados para la transmisión al dispositivo de destino 14. En algunos ejemplos, el dispositivo fuente 12 y el dispositivo de destino 14 operan de manera sustancialmente

simétrica, de modo que cada dispositivo fuente 12 y dispositivo de destino 14 incluye componentes de codificación y decodificación de video. Por lo tanto, el sistema 10 puede permitir la transmisión de video unidireccional o bidireccional entre el dispositivo fuente 12 y el dispositivo de destino 14, por ejemplo, para *streaming* de video, repetición de video, transmisión de video o videotelefonía.

[0036] En el ejemplo de la Figura 1, el dispositivo fuente 12 incluye la fuente de video 18, los medios de almacenamiento 19 configurados para almacenar datos de video, el codificador de video 20 y la interfaz de salida 22. El dispositivo de destino 14 incluye la interfaz de entrada 26, los medios de almacenamiento 28 configurados para almacenar datos de video codificados, el decodificador de video 30, y el dispositivo de visualización 32. En otros ejemplos, el dispositivo fuente 12 y el dispositivo de destino 14 incluyen otros componentes o disposiciones. Por ejemplo, el dispositivo fuente 12 puede recibir datos de video de una fuente de video externa, tal como una cámara externa. Asimismo, el dispositivo de destino 14 puede interactuar con un dispositivo externo de visualización, en lugar de incluir un dispositivo integrado de visualización.

[0037] La fuente de video 18 es una fuente de datos de video. Los datos de video pueden comprender una serie de imágenes. La fuente de video 18 puede incluir un dispositivo de captura de video, tal como una cámara de video, un archivo de video que contiene video capturado anteriormente y/o una interfaz de transmisión de video para recibir datos de video de un proveedor de contenidos de video. En algunos ejemplos, la fuente de video 18 genera datos de video basados en gráficos informáticos o una combinación de video en vivo, video archivado y video generado por computadora.- Los medios de almacenamiento 19 se pueden configurar para almacenar los datos de video. En cada caso, el video capturado, precapturado o generado por computadora puede ser codificado por el codificador de video 20.

[0038] La interfaz de salida 22 puede enviar la información de video codificada al medio legible por computadora 16. La interfaz de salida 22 puede comprender diversos tipos de componentes o dispositivos. Por ejemplo, la interfaz de salida 22 puede comprender un transmisor inalámbrico, un módem, un componente de red alámbrica (por ejemplo, una tarjeta Ethernet) u otro componente físico. En aquellos ejemplos en donde la interfaz de salida 22 comprende un transmisor inalámbrico, la interfaz de salida 22 se puede configurar para transmitir datos, tales como datos de video codificados, modulados de acuerdo con un estándar de comunicación celular, tal como 4G, 4G-LTE, LTE Advanced, 5G y similares. En algunos ejemplos en donde la interfaz de salida 22 comprende un transmisor inalámbrico, la interfaz de salida 22 se

puede configurar para transmitir datos, tales como datos de video codificados, modulados de acuerdo con otros estándares inalámbricos, tales como la especificación de IEEE 802.11, la especificación de IEEE 802.15 (por ejemplo, ZigBee™), un estándar de Bluetooth™ y similares. Por lo tanto, en algunos ejemplos, el dispositivo fuente 12 comprende un dispositivo de comunicación inalámbrica que incluye un transmisor configurado para transmitir datos de video codificados. En algunos de dichos ejemplos, el dispositivo de comunicación inalámbrica comprende un microteléfono y el transmisor está configurado para modular, de acuerdo con un estándar de comunicación inalámbrica, una señal que comprende los datos de video codificados.

[0039] En algunos ejemplos, el circuito de la interfaz de salida 22 se integra al circuito del codificador de video 20 y/u otros componentes del dispositivo fuente 12. Por ejemplo, el codificador de video 20 y la interfaz de salida 22 pueden ser partes de un sistema en un chip (SoC). El SoC también puede incluir otros componentes, tales como un microprocesador de propósito general, una unidad de procesamiento gráfico, etcétera.

[0040] El dispositivo de destino 14 puede recibir los datos de video codificados que se desean decodificar mediante el medio legible por computadora 16. El medio legible por computadora 16 puede comprender cualquier tipo de medio o dispositivo que puede mover los datos de video codificados del dispositivo fuente 12 al dispositivo de destino 14. En algunos ejemplos, el medio legible por computadora 16 comprende un medio de comunicación que permite que el dispositivo fuente 12 transmita los datos de video codificados directamente al dispositivo de destino 14 en tiempo real. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrica o alámbrica, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global, tal como Internet. El medio de comunicación puede incluir *routers*, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo fuente 12 al dispositivo de destino 14. El dispositivo de destino 14 puede comprender uno o más medios de almacenamiento de datos configurados para almacenar datos de video codificados y datos de video decodificados.

[0041] En algunos ejemplos, la interfaz de salida 22 puede enviar datos, tales como datos de video codificados, a un dispositivo intermediario, tal como un dispositivo de

almacenamiento. De manera similar, la interfaz de entrada 26 del dispositivo de destino 14 puede recibir datos codificados del dispositivo intermediario. El dispositivo intermediario puede incluir cualquiera de los diversos medios de almacenamiento de datos distribuidos o que se pueden acceder localmente, tales como un disco duro, discos Blu-ray, DVD, CD-ROM, memoria flash, memoria volátil o no volátil, o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de video codificados. En algunos ejemplos, el dispositivo intermediario corresponde a un servidor de archivos. Los servidores de archivos de ejemplo incluyen servidores web, servidores FTP, dispositivos de almacenamiento conectado en red (NAS) o unidades de disco local.

[0042] El dispositivo de destino 14 puede acceder a los datos de video codificados a través de cualquier conexión de datos estándar, que incluye una conexión a Internet. Esto puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión alámbrica (por ejemplo, DSL, cable módem, etc.) o una combinación de ambos que sea adecuada para acceder a datos de video codificados almacenados en un servidor de archivos. La transmisión de datos de video codificados del dispositivo de almacenamiento puede ser una transmisión por *streaming*, una transmisión por descarga o una combinación de las dos.

[0043] El medio legible por computadora 16 puede incluir medios transitorios, tales como una transmisión de red inalámbrica o alámbrica, o medios de almacenamiento (es decir, medios de almacenamiento no transitorios), tales como disco duro, unidad de disco flash, disco compacto, disco de video digital, disco Blu-ray, u otros medios legibles por computadora. En algunos ejemplos, un servidor de red (no se muestra) puede recibir datos de video codificados del dispositivo fuente 12 y proporcionar los datos de video codificados al dispositivo de destino 14, por ejemplo, mediante transmisión de red. De manera similar, un dispositivo informático de una instalación de producción de medios, tal como una instalación para estampar discos, puede recibir datos de video codificados del dispositivo fuente 12 y producir un disco que contiene los datos de video codificados. Por lo tanto, el medio legible por computadora 16 puede incluir uno o más medios legibles por computadora de diversas formas, en diversos ejemplos.

[0044] La interfaz de entrada 26 del dispositivo de destino 14 recibe datos del medio legible por computadora 16. La interfaz de entrada 26 puede comprender diversos tipos de componentes o dispositivos. Por ejemplo, la interfaz de entrada 26 puede comprender un receptor inalámbrico, un módem, un componente de red alámbrica (por

ejemplo, una tarjeta Ethernet) u otro componente físico. En aquellos ejemplos en donde la interfaz de entrada 26 comprende un receptor inalámbrico, la interfaz de entrada 26 se puede configurar para recibir datos, tales como la corriente de bits, modulados de acuerdo con un estándar de comunicación celular, tal como 4G, 4G-LTE, LTE Advanced, 5G y similares. En algunos ejemplos en donde la interfaz de entrada 26 comprende un receptor inalámbrico, la interfaz de entrada 26 se puede configurar para recibir datos, tales como la corriente de bits, modulados de acuerdo con otros estándares inalámbricos, tales como la especificación de IEEE 802.11, la especificación de IEEE 802.15 (por ejemplo, ZigBee TM), un estándar de Bluetooth TM y similares. Por lo tanto, en algunos ejemplos, el dispositivo de destino 14 puede comprender un dispositivo de comunicación inalámbrica que comprende un receptor configurado para recibir datos de video codificados. En algunos de dichos ejemplos, el dispositivo de comunicación inalámbrica comprende un microteléfono y el receptor está configurado para demodular, de acuerdo con un estándar de comunicación inalámbrica, una señal que comprende los datos de video codificados. En algunos ejemplos, el dispositivo fuente 12 puede comprender un transmisor y el dispositivo de destino 14 puede comprender un transmisor y un receptor.

[0045] En algunos ejemplos, el circuito de la interfaz de entrada 26 se puede integrar al circuito del decodificador de video 30 y/u otros componentes del dispositivo de destino 14. Por ejemplo, el decodificador de video 30 y la interfaz de entrada 26 pueden ser partes de un SoC. El SoC también puede incluir otros componentes, tales como un microprocesador de propósito general, una unidad de procesamiento gráfico, etcétera.

[0046] Los medios de almacenamiento 28 se pueden configurar para almacenar datos de video codificados, tales como datos de video codificados (por ejemplo, una corriente de bits) recibidos mediante la interfaz de entrada 26. El dispositivo de visualización 32 muestra los datos de video decodificados a un usuario. El dispositivo de visualización 32 puede comprender cualquiera de los diversos dispositivos de visualización, tales como 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.

[0047] El codificador de video 20 y el decodificador de video 30 se pueden implementar, cada uno, como cualquiera de los diversos circuitos programables y/o de función fija adecuados, tales como uno o más microprocesadores, procesadores de señal digital (DSP), circuitos integrados para aplicaciones específicas (ASIC), matrices

de compuerta programable en campo (FPGA), lógica discreta, *software*, *hardware*, *firmware* o cualquier combinación de estos. Cuando las técnicas se implementan de manera parcial en el *software*, un dispositivo puede almacenar instrucciones para el *software* es un medio legible por computadora no transitorio adecuado y puede ejecutar las instrucciones en el *hardware* usando uno o más procesadores para realizar las técnicas de la presente divulgación. El codificador de video 20 y el decodificador de video 30 pueden estar incluidos, cada uno, en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) combinado en un dispositivo respectivo.

[0048] En algunos ejemplos, el codificador de video 20 y el decodificador de video 30 codifican y decodifican datos de video de acuerdo con un estándar de codificación de video o especificación. Por ejemplo, el codificador de video 20 y el decodificador de video 30 pueden codificar y decodificar datos de video de acuerdo con 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 y ITU-T H.264 (también denominado ISO/IEC MPEG-4 AVC), que incluye su extensión de codificación de video escalable (SVC) y extensión de codificación de video multivista (MVC) u otro estándar de codificación de video o especificación. En algunos ejemplos, el codificador de video 20 y el decodificador de video 30 codifican y decodifican datos de acuerdo con la codificación de video de alta eficiencia (HEVC), también denominada ITU-T H.265, sus extensiones de codificación de rango y de contenido de pantalla, su extensión de codificación de video 3D (3D-HEVC), su extensión multivista (MV-HEVC) o su extensión escalable (SHVC). En algunos ejemplos, el codificador de video 20 y el decodificador de video 30 operan de acuerdo con otros estándares, que incluyen estándares que actualmente están en desarrollo.

[0049] En HEVC y otras especificaciones de codificación de video, los datos de video incluyen una serie de imágenes. Las imágenes también se pueden denominar “cuadros”. Una imagen puede incluir una o más matrices de muestra. Cada matriz de muestra respectiva de una imagen puede comprender una matriz de muestras para un componente de color respectivo. Una imagen puede incluir tres matices de muestra, indicadas como S_L , S_{Cb} , y S_{Cr} . S_L es una matriz de dos dimensiones (por ejemplo, un bloque) de muestras de luminancia. S_{Cb} es una matriz de dos dimensiones de muestras de crominancia Cb. S_{Cr} es una matriz de dos dimensiones de muestras de crominancia Cr. En otros ejemplos, una imagen puede ser monocromática y solo

puede incluir una matriz de muestras de luminancia.

[0050] Como parte de la codificación de datos de video, un codificador de video 20 puede codificar imágenes de los datos de video. En otras palabras, el codificador de video 20 puede generar representaciones codificadas de las imágenes de los datos de video. Una representación codificada de una imagen puede denominarse en la presente una “imagen codificada”.

[0051] Para generar una representación codificada de una imagen, el codificador de video 20 puede codificar bloques de una imagen. El codificador de video 20 puede incluir, en una corriente de bits, una representación codificada del bloque de video. En algunos ejemplos, para codificar un bloque de la imagen, el codificador de video 20 realiza intrapredicción o interpredicción para generar uno o más bloques predictivos. Asimismo, el codificador de video 20 puede generar datos residuales para el bloque. El bloque residual comprende muestras residuales. Cada muestra residual puede indicar una diferencia entre una muestra de uno de los bloques predictivos generados y una muestra correspondiente del bloque. El codificador de video 20 puede aplicar una transformada a los bloques de muestras residuales para generar coeficientes de la transformada. Además, el codificador de video 20 puede cuantificar los coeficientes de la transformada. En algunos ejemplos, el codificador de video 20 puede generar uno o más elementos sintácticos para representar un coeficiente de la transformada. El codificador de video 20 puede codificar por entropía uno o más de los elementos sintácticos que representan el coeficiente de la transformada.

[0052] De forma más específica, al codificar datos de video de acuerdo con HEVC u otras especificaciones de codificación de video, para generar una representación codificada de una imagen, el codificador de video 20 puede particionar cada matriz de muestra de la imagen en bloques de codificación en árbol (CTB) y codificar los CTB. Un CTB puede ser un bloque de muestras NxN en una matriz de muestra de una imagen. En el perfil *main* de HEVC, el tamaño de un CTB puede variar de 16x16 a 64x64, aunque, técnicamente, se pueden permitir tamaños de CTB de 8x8.

[0053] Una unidad de codificación en árbol (CTU) de una imagen puede comprender uno o más CTB y puede comprender estructuras sintácticas usadas para codificar las muestras del uno o más CTB. Por ejemplo, cada CTU puede comprender un CTB de muestras de luminancia, dos CTB correspondientes de muestras de crominancia y estructuras sintácticas que se usan para codificar las muestras de los CTB. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una CTU puede comprender un solo CTB y estructuras sintácticas que se usan para

codificar las muestras de los CTB. Una CTU también puede denominarse “bloque de árbol” o “unidad de codificación más grande” (LCU). En la presente divulgación, una “estructura sintáctica” se puede definir como cero o más elementos sintácticos que están presentes juntos en una corriente de bits en un orden específico. En algunos códecs, una imagen codificada es una representación codificada que contiene todas las CTU de la imagen.

[0054] Para codificar una CTU de una imagen, el codificador de video 20 puede particionar los CTB de la CTU en uno o más bloques de codificación. Un bloque de codificación es un bloque de muestras $N \times N$. En algunos códecs, para codificar una CTU de una imagen, el codificador de video 20 puede realizar recursivamente la partición del árbol cuaternario en los bloques de codificación en árbol de una CTU para particionar los CTB en bloques de codificación, de aquí el nombre “unidades de codificación en árbol”. Una unidad de codificación (CU) puede comprender uno o más bloques de codificación y estructuras sintácticas usadas para codificar las muestras del uno o más bloques de codificación. Por ejemplo, una CU puede comprender un bloque de codificación de muestras de luminancia y dos bloques de codificación correspondientes de muestras de crominancia de una imagen que tiene una matriz de muestras de luminancia, una matriz de muestras de Cb y una matriz de muestras de Cr, y estructuras sintácticas que se usan para codificar las muestras de los bloques de codificación. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una CU puede comprender un solo bloque de codificación y las estructuras sintácticas que se usan para codificar las muestras del bloque de codificación.

[0055] Además, el codificador de video 20 puede codificar las CU de una imagen de los datos de video. En algunos códecs, como parte de la codificación de una CU, el codificador de video 20 puede particionar un bloque de codificación de la CU en uno o más bloques de predicción. Un bloque de predicción es un bloque rectangular (es decir, cuadrado o no cuadrado) de muestras sobre las cuales se aplica la misma predicción. Una unidad de predicción (PU) de una CU puede comprender uno o más bloques de predicción de una CU y estructuras sintácticas que se usan para predecir el uno o más bloques de predicción. Por ejemplo, una PU puede comprender un bloque de predicción de muestras de luminancia, dos bloques de predicción correspondientes de muestras de crominancia y estructuras sintácticas que se usan para predecir los bloques de predicción. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una PU puede comprender un solo bloque de predicción y estructuras sintácticas que se usan para predecir el bloque de

predicción.

[0056] El codificador de video 20 puede generar un bloque predictivo (por ejemplo, un bloque predictivo de luminancia, Cb y Cr) para un bloque de predicción (por ejemplo, un bloque predicción de luminancia, Cb y Cr) de una PU de una CU. El codificador de video 20 puede usar intrapredicción o interpredicción para generar un bloque predictivo. Si el codificador de video 20 utiliza la intrapredicción para generar un bloque predictivo, el codificador de video 20 puede generar el bloque predictivo en función de muestras decodificadas de la imagen que incluye la CU. Si el codificador de video 20 utiliza la interpredicción para generar un bloque predictivo de una PU de una imagen actual, el codificador de video 20 puede generar el bloque predictivo de la PU en función de muestras decodificadas de una imagen de referencia (es decir, una imagen distinta de la imagen actual). En HEVC, el codificador de video 20 genera una estructura sintáctica de “prediction_unit” dentro de una estructura sintáctica de “coding_unit” para las PU interpredicadas, pero no genera una estructura sintáctica de “prediction_unit” dentro de una estructura sintáctica de “coding_unit” para las PU intrapredicadas. En su lugar, en HEVC, los elementos sintácticos relacionados con las PU intrapredicadas se incluyen directamente en la estructura sintáctica de “coding_unit”.

[0057] El codificador de video 20 puede generar uno o más bloques residuales para una CU. Por ejemplo, el codificador de video 20 puede generar un bloque residual de luminancia para la CU. Cada muestra en el bloque residual de luminancia de la CU indica una diferencia entre una muestra de luminancia en uno de los bloques de luminancia predictivos de la CU y una muestra correspondiente en el bloque de codificación de luminancia original de la CU. Asimismo, el codificador de video 20 puede generar un bloque residual de Cb para la CU. Cada muestra en el bloque residual de Cb de una CU puede indicar una diferencia entre una muestra de Cb en uno de los bloques de Cb predictivos de la CU y una muestra correspondiente en el bloque de codificación de Cb original de la CU. El codificador de video 20 también puede generar un bloque residual de Cr para la CU. Cada muestra en el bloque residual de Cr de la CU puede indicar una diferencia entre una muestra de Cr en uno de los bloques de Cr predictivos de la CU y una muestra correspondiente en el bloque de codificación de Cr original de la CU.

[0058] Además, el codificador de video 20 puede descomponer los bloques residuales de una CU en uno o más bloques de la transformada. Por ejemplo, el codificador de video 20 puede usar la partición del árbol cuaternario para descomponer los bloques

residuales de una CU en uno o más bloques de la transformada. Un bloque de la transformada es un bloque rectangular (por ejemplo, cuadrado o no cuadrado) de muestras sobre las cuales se aplica la misma transformada. Una unidad de la transformada (TU) de una CU puede comprender uno o más bloques de la transformada. Por ejemplo, una TU puede comprender un bloque de la transformada de muestras de luminancia, dos bloques de la transformada correspondientes de muestras de crominancia y estructuras sintácticas que se usan para transformar las muestras del bloque de la transformada. Por lo tanto, cada TU de una CU puede tener un bloque de la transformada de luminancia, un bloque de la transformada de Cb y un bloque de la transformada de Cr. El bloque de la transformada de luminancia de la TU puede ser un subbloque del bloque residual de luminancia de la CU. El bloque de la transformada de Cb puede ser un subbloque del bloque residual de Cb de la CU. El bloque de la transformada de Cr puede ser un subbloque del bloque residual de Cr de la CU. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una TU puede comprender un solo bloque de la transformada y las estructuras sintácticas que se usan para transformar las muestras del bloque de la transformada.

[0059] El codificador de video 20 puede aplicar una o más transformadas a un bloque de la transformada de una TU para generar un bloque de coeficiente para la TU. Un bloque de coeficiente puede ser una matriz de coeficientes de la transformada bidimensional. Un coeficiente de la transformada puede ser una cantidad escalar. En algunos ejemplos, la una o más transformadas convierten el bloque de la transformada de un dominio de píxeles a un dominio de frecuencia. Por lo tanto, en tales ejemplos, un coeficiente de la transformada puede ser una cantidad escalar que se considera que está en un dominio de frecuencia. Un nivel del coeficiente de la transformada es una cantidad entera que representa un valor asociado a un índice de frecuencia de dos dimensiones particular en un proceso de decodificación anterior al ajuste para el cálculo de un valor del coeficiente de la transformada.

[0060] En algunos ejemplos, el codificador de video 20 salta la aplicación de las transformadas al bloque de la transformada. En tales ejemplos, el codificador de video 20 puede tratar los valores de las muestras residuales de la misma forma que los coeficientes de la transformada. Por lo tanto, en aquellos ejemplos en los que el codificador de video 20 salta la aplicación de las transformadas, la siguiente discusión sobre coeficientes de la transformada y bloques de coeficiente puede ser aplicable a los bloques de la transformada de muestras residuales.

[0061] Después de generar un bloque de coeficiente, el codificador de video 20 puede cuantificar el bloque de coeficiente para reducir posiblemente la cantidad de datos usados para representar el bloque de coeficiente, lo que proporciona una mayor compresión de forma potencial. En general, la cuantificación se refiere al proceso en el que se comprime un rango de valores a un solo valor. Por ejemplo, la cuantificación se puede llevar a cabo al dividir un valor por una constante y luego redondear en el entero más cercano. Para cuantificar el bloque de coeficiente, el codificador de video 20 puede cuantificar los coeficientes de la transformada del bloque de coeficiente. En algunos ejemplos, el codificador de video 20 salta la cuantificación.

[0062] El codificador de video 20 puede generar elementos sintácticos que indican algunos o todos los coeficientes de la transformada cuantificados de forma potencial. El codificador de video 20 puede codificar por entropía uno o más de los elementos sintácticos que indican un coeficiente de la transformada cuantificado. Por ejemplo, el codificador de video 20 puede realizar la codificación aritmética binaria adaptativa según el contexto (CABAC) en los elementos sintácticos que indican los coeficientes de la transformada cuantificados. Por lo tanto, un bloque codificado (por ejemplo, una CU codificada) puede incluir los elementos sintácticos codificados por entropía que indican los coeficientes de la transformada cuantificados.

[0063] El codificador de video 20 puede enviar una corriente de bits que incluye datos de video codificados. En otras palabras, el codificador de video 20 puede generar una corriente de bits que incluye una representación codificada de los datos de video. La representación codificada de los datos de video puede incluir una representación codificada de imágenes de los datos de video. Por ejemplo, la corriente de bits puede comprender una secuencia de bits que forma una representación de imágenes codificadas de los datos de video y datos asociados. En algunos ejemplos, una representación de una imagen codificada puede incluir representaciones codificadas de bloques de la imagen.

[0064] Una corriente de bits puede comprender una secuencia de unidades de capa de abstracción de red (NAL). Una unidad de NAL es una estructura sintáctica que contiene una indicación del tipo de datos en la unidad de NAL y bytes que contienen esos datos en forma de una carga útil de secuencia de bytes en bruto (RBSP) intercalada según sea necesario con bits de prevención de emulación. Cada una de las unidades de NAL puede incluir un encabezado de la unidad de NAL y puede encapsular una RBSP. El encabezado de la unidad de NAL puede incluir un elemento sintáctico que indica un código de tipo de unidad de NAL. El código de tipo de unidad

de NAL especificado por el encabezado de la unidad de NAL de una unidad de NAL indica el tipo de la unidad de NAL. Una RBSP puede ser una unidad sintáctica que contiene un número entero de bytes que está encapsulado dentro de una unidad de NAL. En algunos ejemplos, una RBSP no incluye ningún bit.

[0065] El decodificador de video 30 puede recibir una corriente de bits generada por el codificador de video 20. Como se indicó anteriormente, la corriente de bits puede comprender una representación codificada de los datos de video. El decodificador de video 30 puede decodificar la corriente de bits para reconstruir las imágenes de los datos de video. Como parte de la decodificación de la corriente de bits, el decodificador de video 30 puede obtener elementos sintácticos de la corriente de bits. El decodificador de video 30 puede reconstruir las imágenes de los datos de video en función de, al menos en parte, los elementos sintácticos obtenidos de la corriente de bits. Generalmente, el proceso para reconstruir las imágenes de los datos de video es recíproco al proceso realizado por el codificador de video 20 para codificar las imágenes.

[0066] Por ejemplo, como parte de la decodificación de una imagen de los datos de video, el decodificador de video 30 puede usar interpretación o intrapredicción para generar bloques predictivos. De forma adicional, el decodificador de video 30 puede determinar los coeficientes de la transformada en función de los elementos sintácticos obtenidos de la corriente de bits. En algunos ejemplos, el decodificador de video 30 cuantifica de manera inversa los coeficientes de la transformada determinados. Además, el decodificador de video 30 puede aplicar una transformada inversa a los coeficientes de la transformada determinados para determinar los valores de las muestras residuales. El decodificador de video 30 puede reconstruir un bloque de la imagen en función de las muestras residuales y las muestras correspondientes de los bloques predictivos generados. Por ejemplo, el decodificador de video 30 puede agregar muestras residuales a las muestras correspondientes de los bloques predictivos generados para determinar las muestras reconstruidas del bloque.

[0067] De forma más específica, en HEVC y otras especificaciones de codificación de video, el decodificador de video 30 puede usar interpretación o intrapredicción para generar uno o más bloques predictivos para cada PU de una CU actual. Además, el decodificador de video 30 puede cuantificar de manera inversa los bloques de coeficiente de las TU de la CU actual. El decodificador de video 30 puede realizar transformadas inversas en los bloques de coeficiente para reconstruir bloques de la transformada de las TU de la CU actual. El decodificador de video 30 puede

reconstruir un bloque de codificación de la CU actual en función de las muestras de los bloques predictivos de las PU de la CU actual y las muestras residuales de los bloques de la transformada de las TU de la CU actual. En algunos ejemplos, el decodificador de video 30 puede reconstruir los bloques de codificación de la CU actual agregando las muestras de los bloques predictivos para las PU de la CU actual a las muestras decodificadas correspondientes de los bloques de la transformada de las TU de la CU actual. Al reconstruir los bloques de codificación para cada CU de una imagen, el decodificador de video 30 puede reconstruir la imagen.

[0068] Como se mencionó anteriormente, un codificador de video (por ejemplo, el codificador de video 20 o el decodificador de video 30) puede aplicar la interpredicción para generar un bloque predictivo para un bloque de video de una imagen actual. Por ejemplo, el codificador de video puede aplicar la interpredicción para generar un bloque de predicción de una CU. Si el codificador de video aplica la interpredicción para generar un bloque de predicción, el codificador de video genera el bloque de predicción en función de las muestras decodificadas de una o más imágenes de referencia. Generalmente, las imágenes de referencia son imágenes distintas a la imagen actual. En algunas especificaciones de codificación de video, un codificador de video también puede tratar a la imagen actual misma como una imagen de referencia.

[0069] Si un codificador de video (por ejemplo, el codificador de video 20 o el decodificador de video 30) comienza a procesar una imagen actual, el codificador de video puede determinar uno o más subconjuntos de conjunto de imágenes de referencia (RPS) para la imagen actual. Por ejemplo, en HEVC, un codificador de video puede determinar los siguientes subconjuntos de RPS: RefPicSetStCurrBefore, RefPicSetStCurrAfter, RefPicSetFoll, RefPicSetLtCurr y RefPicSetLtFoll. Además, el codificador de video puede determinar una o más listas de imágenes de referencia. Cada una de las listas de imágenes de referencia de una imagen actual incluye cero o más imágenes de referencia del RPS de la imagen actual. Una de las listas de imágenes de referencia se puede denominar Lista de Imágenes de Referencia 0 (RefPicList0) y otra lista de imágenes de referencia se puede denominar Lista de Imágenes de Referencia 1 (RefPicList1).

[0070] Un segmento de una imagen puede incluir un número entero de bloques de la imagen. Por ejemplo, en HEVC y otras especificaciones de codificación de video, un segmento de una imagen puede incluir un número entero de las CTU de la imagen. Las CTU de un segmento se pueden ordenar de forma consecutiva en un orden de exploración, tal como un orden de exploración de trama. En HEVC y otros estándares

de codificación de video, un segmento se define como un número entero de las CTU contenidas en un segmento independiente y todos los segmentos dependientes posteriores (de haberlos) que preceden al próximo segmento independiente (de haberlo) dentro de la misma unidad de acceso. Además, en HEVC y otros estándares de codificación de video, un segmento se define como un número entero de las CTU ordenadas de forma consecutiva en la exploración de mosaicos y contenidas en sola unidad de NAL. Una exploración de mosaicos es un orden secuencial específico de los CTB que particionan una imagen en el que los CTB se ordenan de forma consecutiva en una exploración en trama de CTB en un mosaico, mientras que los mosaicos en una imagen se ordenan de forma consecutiva en una exploración en trama de los mosaicos de la imagen. Un mosaico es una región rectangular de los CTB dentro de una columna de mosaicos particular y una fila de mosaicos particular en una imagen.

[0071] Como se indicó anteriormente, una corriente de bits puede incluir una representación de imágenes codificadas de los datos de video y datos asociados. Los datos asociados pueden incluir conjuntos de parámetros. Las unidades de NAL pueden encapsular las RBSP para conjuntos de parámetros de video (VPS), conjuntos de parámetros de secuencias (SPS) y conjuntos de parámetros de imágenes (PPS). Un VPS es una estructura sintáctica que comprende elementos sintácticos que se aplican a cero o más secuencias enteras de video codificadas (CVS). Un SPS también es una estructura sintáctica que comprende elementos sintácticos que se aplican a cero o más CVS enteras. Un SPS puede incluir un elemento sintáctico que identifica un VPS que está activo cuando el SPS está activo. Por lo tanto, los elementos sintácticos de un VPS pueden ser, en general, más aplicables que los elementos sintácticos de un SPS. Un PPS es una estructura sintáctica que comprende elementos sintácticos que se aplican a cero o más imágenes codificadas. Un PPS puede incluir un elemento sintáctico que identifica un SPS que está activo cuando el PPS está activo. Un encabezado de segmento de un segmento puede incluir un elemento sintáctico que indica un PPS que está activo cuando el segmento está siendo codificado.

[0072] Como se analizó anteriormente, un codificador de video puede generar una corriente de bits que comprende una serie de unidades de NAL. En la codificación de video multicapa, las diferentes unidades de NAL de la corriente de bits se pueden asociar a diferentes capas de la corriente de bits. Una capa se puede definir como un conjunto de unidades de NAL de VCL y unidades de NAL que no son de VCL asociadas que tienen el mismo identificador de capas. Una capa puede ser igual a una

vista en la codificación de video multivista. En la codificación de video multivista, una capa puede contener todos los componentes de vista de la misma capa con diferentes marcas temporales. Cada componente de vista puede ser una imagen codificada de la escena de video que pertenece a una vista específica en una marca temporal específica. En la codificación de video multicapa, la expresión “unidad de acceso” se puede referir a un conjunto de imágenes que corresponden a la misma marca temporal. Por lo tanto, un “componente de vista” puede ser una representación codificada de una vista en una sola unidad de acceso. En algunos ejemplos, un componente de vista puede comprender un componente de vista de textura (es decir, una imagen de textura) o un componente de vista de profundidad (es decir, una imagen de profundidad).

[0073] En algunos ejemplos de codificación de video multivista, una capa puede contener todas las imágenes de profundidad codificadas de una vista específica o imágenes de textura codificadas de una vista específica. En otros ejemplos de codificación de video multivista, una capa puede contener componentes de vista de textura y componentes de vista de profundidad de una vista específica. De forma similar, en el contexto de la codificación de video escalable, en general, una capa corresponde a imágenes codificadas que tienen características de video diferentes a las imágenes codificadas en otras capas. Generalmente, estas características de video incluyen resolución espacial y nivel de calidad (por ejemplo, relación señal/ruido).

[0074] Para cada capa respectiva de corriente de bits, los datos en una capa inferior se pueden decodificar sin hacer referencia a los datos en cualquier capa superior. En la codificación de video escalable, por ejemplo, los datos en una capa base se pueden decodificar sin hacer referencia a los datos en una capa de mejora. En general, las unidades de NAL solo pueden encapsular los datos de una sola capa. Por lo tanto, las unidades de NAL que encapsulan los datos de la capa restante más alta de la corriente de bits se pueden eliminar de la corriente de bits sin afectar la capacidad de decodificación de los datos en las capas restantes de la corriente de bits. En la codificación multivista, las capas más altas pueden incluir componentes de vista adicionales. En SHVC, las capas más altas pueden incluir datos de mejora, datos de mejora espacial y/o datos de mejora temporal de la relación señal/ruido (SNR). En MV-HEVC y SHVC, una capa se puede denominar “capa base” si un decodificador de video puede decodificar imágenes en la capa sin hacer referencia a los datos de cualquier otra capa. La capa base puede adecuarse a la especificación base de HEVC (por ejemplo, Rec. ITU-T H.265 | ISO/IEC 23008-2).

[0075] En la codificación de video escalable, las capas distintas a la capa base se pueden denominar “capas de mejora” y pueden proporcionar información que mejora la calidad visual de los datos de video decodificados de la corriente de bits. La codificación de video escalable puede mejorar la resolución espacial, la relación señal/ruido (es decir, la calidad) o la tasa temporal.

[0076] La codificación multivista puede ser compatible con la predicción entrevista. La predicción entrevista es similar a la interpredicción usada en HEVC y puede usar los mismos elementos sintácticos. Sin embargo, si un codificador de video lleva a cabo la predicción entrevista en una unidad de video actual (tal como una PU), el codificador de video 20 puede usar, como imagen de referencia, una imagen que esté en la misma unidad de acceso que la unidad de video actual, pero en una vista diferente. En cambio, la interpredicción convencional solo usa imágenes en unidades de acceso diferentes como imágenes de referencia.

[0077] En la codificación multivista, una vista se puede denominar “vista base” si un decodificador de video (por ejemplo, el decodificador de video 30) puede decodificar imágenes en la vista sin hacer referencia a las imágenes en cualquier otra vista. Al codificar una imagen en una de las vistas que no son de base, un codificador de video (tal como el codificador de video 20 o el decodificador de video 30) puede agregar una imagen a una lista de imágenes de referencia si la imagen se encuentra en una vista diferente pero dentro de una misma marca temporal (es decir, unidad de acceso) que la imagen que el codificador de video está codificando actualmente. Como otras imágenes de referencia de interpredicción, el codificador de video puede insertar una imagen de referencia de predicción entrevista en cualquier posición de una lista de imágenes de referencia.

[0078] Por ejemplo, las unidades de NAL pueden incluir encabezados (es decir, encabezados de unidades de NAL) y cargas útiles (por ejemplo, las RBSP). Los encabezados de unidades de NAL pueden incluir elementos sintácticos de identificador de capas (por ejemplo, elementos sintácticos `nuh_layer_id` en HEVC). Las unidades de NAL que tienen elementos sintácticos de identificador de capas que especifican diferentes valores pertenecen a diferentes “capas” de una corriente de bits. Por lo tanto, en la codificación de vista multicapa (por ejemplo, MV-HEVC, SVC o SHVC), el elemento sintáctico de identificador de capas de la unidad de NAL especifica un identificador de capas (es decir, un ID de capa) de la unidad de NAL. El identificador de capas de una unidad de NAL es igual a 0 si la unidad de NAL se relaciona con una capa base en la codificación multicapa. Los datos en una capa base

de una corriente de bits se pueden decodificar sin hacer referencia a los datos en cualquier otra capa de la corriente de bits. Si la unidad de NAL no se relaciona con una capa base en la codificación multicapa, el identificador de capas de la unidad de NAL puede tener un valor que no sea cero. En la codificación multivista, las diferentes capas de una corriente de bits pueden corresponder a diferentes vistas. En la codificación de video escalable (por ejemplo, SVC o SHVC), las capas distintas a la capa base se pueden denominar “capas de mejora” y pueden proporcionar información que mejora la calidad visual de los datos de video decodificados de la corriente de bits.

[0079] Además, algunas imágenes dentro de una capa se pueden decodificar sin hacer referencia a otras imágenes dentro de la misma capa. Por lo tanto, las unidades de NAL que encapsulan los datos de ciertas imágenes de una capa se pueden eliminar de la corriente de bits sin afectar la capacidad de decodificación de otras imágenes en la capa. La eliminación de las unidades de NAL que encapsulan los datos de tales imágenes puede reducir la velocidad de cuadro de la corriente de bits. Un subconjunto de imágenes dentro de una capa que se puede decodificar sin hacer referencia a otras imágenes dentro de la capa se puede denominar “subcapa”, “capa temporal” o una “subcapa temporal”. La capa temporal superior puede incluir todas las imágenes en la capa. Por lo tanto, la escalabilidad temporal se puede alcanzar dentro de una capa al definir un grupo de imágenes con un nivel temporal particular como una subcapa (es decir, capa temporal).

[0080] Las unidades de NAL pueden incluir elementos sintácticos de identificador temporal (por ejemplo, temporal_id en HEVC). El elemento sintáctico de identificador temporal de una unidad de NAL especifica un identificador temporal de la unidad de NAL. El identificador temporal de una unidad de NAL identifica una subcapa temporal con la que se asocia la unidad de NAL. Por lo tanto, cada subcapa temporal de una corriente de bits se puede asociar a un identificador temporal diferente. Si el identificador temporal de una primera unidad de NAL es inferior al identificador temporal de una segunda unidad de NAL, los datos encapsulados por la primera unidad de NAL se pueden decodificar sin hacer referencia a los datos encapsulados por la segunda unidad de NAL.

[0081] 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 y ITU-T H.264 (también denominado ISO/IEC MPEG-4 AVC), que incluye su extensión de codificación de video escalable (SVC) y extensión de codificación de video multivista (MVC). Además, un nuevo estándar de codificación de video, a saber

codificación de video de alta eficacia (HEVC), se desarrolló de forma reciente por el Equipo de Colaboración Conjunta sobre Codificación de Video (JCT-VC) de ITU-T del Grupo de Expertos en Codificación de Video (VCEG) y el ISO/IEC Grupo de Expertos en Imágenes en Movimiento (MPEG). Wang et al., “*High Efficiency Video Coding (HEVC) Defect Report*”, Equipo de Colaboración Conjunta sobre Codificación de Video (JCT-VC) de ITU-T SG 16 WP 3 y ISO/IEC JTC 1/SC 29/WG 11, 14.^a Reunión, Viena, AT, 25 de julio – 2 de agosto de 2013, documento JCTVC-N1003-v1, es una especificación borrador de HEVC. El estándar de HEVC se finalizó en enero de 2013.

[0082] ITU-T VCEG (Q6/16) e ISO/IEC MPEG (JTC 1/SC 29/WG 11) están estudiando la necesidad potencial de la estandarización de la tecnología de codificación de video futura con una capacidad de compresión que exceda de forma significativa la del estándar de HEVC actual (que incluye sus extensiones actuales y de corto plazo para la codificación del contenido de pantalla y codificación de alto rango dinámico). Los grupos están trabajando juntos en esta actividad de exploración en un esfuerzo de colaboración conjunta conocido como Equipo de Exploración de Video Conjunta (JVET) para evaluar diseños de tecnología de compresión propuestos por los expertos en esta área. El JVET se reunió por primera vez del 19 al 21 de octubre de 2015. El modelo de exploración conjunta (JEM) es un modelo de prueba realizado por el JVET. J. Chen et al., “*Description of Exploration Experiments on Coding Tools*”, JVET-D1011, Chengdu, octubre de 2016 es una descripción de un algoritmo para la cuarta versión del JEM (es decir, JEM4).

[0083] En el campo de la codificación de video, es común aplicar filtros para mejorar la calidad de la señal de video decodificada. El filtro se puede aplicar como un posfiltro, en donde el cuadro filtrado no se usa para la predicción de cuadros futuros o como un filtro en bucle, en donde el cuadro filtrado se usa para predecir cuadros futuros. Se puede diseñar un filtro, por ejemplo, al minimizar el error entre la señal original y la señal filtrada decodificada. De manera similar a los coeficientes de la transformada, los coeficientes del filtro $h(k, l)$, $k = -K, \dots, K, l = -K, \dots, K$ se pueden cuantificar de la siguiente manera:

$$f(k, l) = \text{round}(\text{normFactor} \cdot h(k, l))$$

y codificar y enviar al decodificador. El *normFactor*, en general, es igual a 2^n . Mientras más grande es el valor del *normFactor*, más precisa es la cuantificación y los coeficientes de filtro cuantificados $f(k, l)$ proporcionan un mayor rendimiento. Por otra parte, los valores más grandes del *normFactor* producen los coeficientes $f(k, l)$ que requieren más bits para transmitir.

[0084] En el decodificador de video 30, los coeficientes de filtro decodificados $f(k, l)$ se aplican a la imagen reconstruida $R(i, j)$ de la siguiente manera:

$$\tilde{R}(i, j) = \sum_{k=-K}^K \sum_{l=-K}^K f(k, l) R(i + k, j + l) / \sum_{k=-K}^K \sum_{l=-K}^K f(k, l), \quad (1)$$

en donde i y j son las coordenadas de los píxeles dentro del cuadro. El filtro adaptativo en bucle se evaluó en la etapa de HEVC, pero no se incluyó en la versión final.

[0085] El filtro adaptativo en bucle empleado en el JEM se describió en J. Chen et al., “Coding tools investigation for next generation video coding”, SG16-Geneva-C806, enero de 2015. La idea básica es la misma que la adaptación con base en ALF con bloques en T. Wiegand et al., “WD3: Working Draft 3 of High-Efficiency Video Coding”, Equipo de Colaboración Conjunta sobre Codificación de Video (JCT-VC) de ITU-T SG16 WP3 y ISO/IEC JTC1/SC29/WG11, JCTVC-E603, 5.^a Reunión: Ginebra, CH, 16-23 de marzo de 2011, en adelante, “JCTVC-E603”.

[0086] Para el componente de luminancia, los bloques 4x4 en la imagen entera se clasifican en función de una dirección laplaciana de una dimensión (hasta 3 direcciones) y una actividad laplaciana de dos dimensiones (hasta 5 valores de actividad). El cálculo de la dirección Dir_b y la actividad sin cuantificar Act_b se muestra en las ecuaciones (2) a (5), en donde $\hat{I}_{i,j}$ indica un píxel reconstruido con coordenadas relativas (i, j) a la parte superior izquierda de un bloque de 4x4. Act_b , además, se cuantifica al rango de 0 a 4, de forma inclusiva, como se describe en JCTVC-E603.

$$V_{i,j} = |\hat{I}_{i,j} \times 2 - \hat{I}_{i,j-1} - \hat{I}_{i,j+1}| \quad (2)$$

$$H_{i,j} = |\hat{I}_{i,j} \times 2 - \hat{I}_{i-1,j} - \hat{I}_{i+1,j}| \quad (3)$$

$$Dir_b = \begin{cases} 1, & \text{if } (\sum_{i=0}^3 \sum_{j=0}^3 H_{i,j} > 2 \times \sum_{i=0}^3 \sum_{j=0}^3 V_{i,j}) \\ 2, & \text{if } (\sum_{i=0}^3 \sum_{j=0}^3 V_{i,j} > 2 \times \sum_{i=0}^3 \sum_{j=0}^3 H_{i,j}) \\ 0, & \text{de otro modo} \end{cases} \quad (4)$$

$$Act_b = \sum_{i=0}^3 \sum_{j=0}^3 (\sum_{m=i-1}^{i+1} \sum_{n=j-1}^{j+1} (V_{m,n} + H_{m,n})) \quad (5)$$

[0087] En total, cada bloque se puede categorizar en uno de los 15 (5x3) grupos y se asigna un índice a cada bloque de 4x4 de acuerdo con el valor de Dir_b y Act_b del bloque. Se indica el índice del grupo con C y se establece que C es igual a $5Dir_b + \hat{A}$ en donde $\hat{A}$ es el valor cuantificado de Act_b . Por lo tanto, el codificador de video 20 puede señalar hasta 15 conjuntos de parámetros de ALF para el componente de luminancia de una imagen. Para ahorrar el costo de la señalización, el codificador de

video 20 puede fusionar los grupos junto con el valor del índice de grupo. Para cada grupo fusionado, el codificador de video 20 puede señalar un conjunto de coeficientes de ALF. La Figura 2 ilustra tres soportes de filtros de ALF de ejemplo diferentes. En el ejemplo de la Figura 2, son compatibles hasta tres formas de filtro simétricas circulares. Para ambos componentes de crominancia en una imagen, se aplica un solo conjunto de coeficientes de ALF y siempre se utiliza el filtro con forma de diamante de 5x5.

[0088] Del lado del decodificador, el decodificador de video 30 puede filtrar cada muestra de píxel $\hat{I}_{i,j}$, lo que da como resultado el valor de píxel $I'_{i,j}$ como se muestra en la ecuación (6), en donde L indica la longitud del filtro, $f_{m,n}$ representa el coeficiente del filtro y o indica la desviación del filtro.

$$I'_{i,j} = \sum_{m=-L}^L \sum_{n=-L}^L f_{m,n} \times \hat{I}_{i+m,j+n} + o \quad (6)$$

En algunos diseños, solo es compatible hasta un filtro para los dos componentes de crominancia.

[0089] A continuación sigue una lista de los datos que se pueden señalar como coeficientes de filtro.

1. **Cantidad total de filtros:** La cantidad total de filtros (o cantidad total de grupos fusionados) se señala en primer lugar cuando ALF se habilita para un segmento. La cantidad total de filtros señalizados se aplica al componente de luminancia. Para los componentes de crominancia, puesto que solo se puede aplicar un filtro de ALF, no se requiere señalar la cantidad total de filtros.
2. **Soporte de filtro:** Se señala un índice de los tres soportes de filtro.
3. **Índice de filtro:** Indica qué filtro de ALF se usa, es decir, la información de fusión de clases. Se pueden fusionar clases que tengan valores no consecutivos de C , es decir, comparten el mismo filtro. Al codificar una etiqueta de cada clase para indicar si la clase se fusiona o no, se puede derivar el índice de filtro. En algunos ejemplos, la información de fusión de clases también se puede señalar para la fusión de un índice de filtro izquierdo o superior.
4. **Etiqueta *forceCoeff0*:** La etiqueta *forceCoeff0* se usa para indicar si al menos uno de los filtros no debe ser codificado. Si esta etiqueta es igual a 0, todos los filtros deben ser codificados. Si la etiqueta *forceCoeff0* es igual a 1, una etiqueta por cada grupo fusionado, indicada como *CodedVarBin*, es, además, señalizada para indicar si el filtro debe ser señalado o no. Si el filtro no es señalado, significa que todos los coeficientes de filtro asociados al filtro son iguales a 0.

5. Método de predicción: Cuando múltiples grupos de filtros requieren ser señalizados, se puede usar uno de los siguientes dos métodos:

- Todos los filtros se codifican directamente en la información del filtro. En este caso, por ejemplo, los valores de los coeficientes de filtro se pueden codificar en la corriente de bits sin usar ninguna técnica de codificación predictiva. En otras palabras, los filtros se señalizan de forma explícita.
- Los coeficientes de filtro de un primer filtro se codifican de forma directa. Mientras que para el resto de los filtros, los coeficientes de filtro se codifican de forma predictiva en la información del filtro. En este caso, los valores de los coeficientes de filtro se pueden definir mediante valores residuales o diferencias con relación a los coeficientes del filtro asociados a un filtro codificado previamente. El filtro codificado previamente es aquel que sea el filtro más reciente (es decir, los índices de filtro del filtro actual y su predictor son consecutivos).

Para indicar el uso de uno de los dos métodos de predicción mencionados anteriormente, el codificador de video 20 puede señalar una etiqueta si la cantidad de grupos fusionados es mayor a 1 y *forceCoeff0* es igual a 0.

[0090] Un conjunto de parámetros de ALF puede incluir uno o más de los elementos sintácticos enumerados anteriormente y también pueden incluir coeficientes de filtro.

[0091] Un codificador de video (por ejemplo, el codificador de video 20 o el decodificador de video 30) también puede usar predicción temporal de los coeficientes de filtro. El codificador de video puede almacenar coeficientes de ALF de imágenes codificadas previamente y puede reutilizar los coeficientes de ALF de las imágenes codificadas previamente como coeficientes de ALF de una imagen actual. El codificador de video 20 puede elegir usar los coeficientes de ALF almacenados para la imagen actual y evitar la señalización de los coeficientes de ALF. En este caso, el codificador de video 20 solo señala un índice para una de las imágenes de referencia (la cual, en realidad, es igual al índice del candidato en la matriz almacenada para parámetros de ALF) y los coeficientes de ALF almacenados de la imagen indicada son simplemente heredados para la imagen actual. Para indicar el uso de predicción temporal, el codificador de video 20 primero puede codificar una etiqueta que indica el uso de predicción temporal antes de enviar el índice a la imagen de referencia.

[0092] En JEM4, los codificadores de video almacenan los parámetros de ALF de, como máximo, 6 imágenes codificadas previamente que se codifican con parámetros

de ALF señalizados (es decir, se inhabilita la predicción temporal) en una matriz aparte. Un codificador de video vacía la matriz de manera eficaz para las imágenes de punto de acceso aleatorio interno (IRAP). Para evitar duplicaciones, el codificador de video únicamente almacena los valores de parámetros de ALF en la matriz si los valores de parámetros de ALF se señalaron de forma explícita. El almacenamiento de los parámetros de ALF se lleva a cabo de acuerdo con el esquema de FIFO, de modo que si la matriz se llena, el codificador de video sobrescribe los valores de parámetros de ALF más antiguos (es decir, parámetros de ALF) con un nuevo conjunto de valores de parámetros de ALF, según un orden de decodificación.

[0093] En M. Karczewicz et al., *“Improvements on adaptive loop filter”*, Equipo de exploración (JVET) de ITU-T SG 16 WP 3 y ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET-B0060_r1, 2.^a Reunión: San Diego, EE.UU., 20–26 de febrero de 2016 (en adelante, “JVET-B0060”), se propone el ALF con base en transformaciones geométricas (GALF). En GALF, la clasificación se modifica teniendo en cuenta gradientes diagonales y se pueden aplicar transformaciones geométricas a los coeficientes de filtro.

[0094] En función de toda la información de gradientes, incluidos gradientes horizontales, verticales y diagonales, se determina una de cuatro transformaciones geométricas de coeficientes de filtro. Es decir, las muestras clasificadas en la misma categoría compartirán los mismos coeficientes de filtro. Sin embargo, la región de soportes de filtros se puede transformar en función del índice de transformación geométrica seleccionado. El método descrito en JVET-B0060 puede reducir de forma eficaz la cantidad de filtros que deben enviarse al decodificador, por lo que se reduce la cantidad de bits que se necesitan para representarlos, o reducir de forma alternativa las diferencias entre los cuadros reconstruidos y los cuadros originales. Cada bloque de 2x2 se clasifica en una de 25 clases en función de su direccionalidad y un valor cuantificado de actividad.

[0095] Además, en JVET-B0060, para mejorar la eficacia de codificación cuando no está disponible la predicción temporal (por ejemplo, en cuadros internos), un codificador de video asigna un conjunto de 16 filtros fijos a cada clase. Es decir, se pueden predefinir 16*25(clases) filtros. Para indicar el uso de un filtro fijo, se señala una etiqueta para cada clase y, de ser necesario, el índice del filtro fijo. Incluso cuando se selecciona el filtro fijo para una clase determinada, aun se pueden enviar los coeficientes del filtro adaptativo $f(k,l)$ para esta clase, en cuyo caso los coeficientes del filtro que se aplicarán a la imagen reconstruida son la suma de ambos conjuntos de coeficientes. Una o más de las clases pueden compartir los mismos coeficientes $f(k,l)$

señalizados en la corriente de bits, incluso si se eligieron diferentes filtros fijos para estas. La publicación de patente estadounidense N.º 2017/0238020, publicada el 17 de agosto de 2017, describe cómo los filtros fijos también se podrían aplicar a cuadros intercodificados.

[0096] En JVET-B0060, el diseño de la predicción temporal de cuadros codificados previamente como en la segunda versión de JEM (es decir, JEM2) se mantiene sin modificaciones. JEM2 se describe en Jianle Chen et al., “*Algorithm Description of Joint Exploration Test Model 2*”, Equipo de Exploración de Video Conjunta (JVET) de ITU-T SG 16 WP 3 y ISO/IEC JTC 1/SC 29/WG 11, 2.ª Reunión, San Diego, EE.UU., 20-26 de febrero de 2016, documento JVET-B1001_v3. Es decir, se codifica una etiqueta para indicar si se usa la predicción temporal de los coeficientes de ALF. Si se usa la predicción temporal de los coeficientes de ALF, además, se señala un índice de las imágenes seleccionadas para los parámetros de ALF almacenados. En este caso, no se requiere señalar los índices de filtro para cada clase y coeficientes de filtro.

[0097] Además, la codificación explícita de los coeficientes de filtro de ALF se puede usar con GALF. Por ejemplo, un patrón de predicción y un índice de predicción de filtros fijos se pueden codificar de forma explícita en GALF. Se definen tres casos:

- caso 1: ninguno de los filtros de las 25 clases se predicen de los filtros fijos;
- caso 2: todos los filtros de las clases se predicen de los filtros fijos; y
- caso 3: los filtros asociados a algunas clases se predicen de filtros fijos y los filtros asociados al resto de las clases no se predicen de los filtros fijos.

Se puede codificar, en primer lugar, un índice para indicar uno de los tres casos. Además, se aplica lo siguiente:

- Si el caso indicado es el caso 1, no se requiere señalar de forma adicional el índice del filtro fijo.
- Por el contrario, si el caso indicado es el caso 2, se señala un índice del filtro fijo seleccionado para cada clase.
- De otro modo, si el caso indicado es el caso 3, se señala, en primer lugar, un bit por cada clase y, si se usa un filtro fijo, se señala de forma adicional el índice del filtro fijo.

[0098] En GALF, para reducir la cantidad de bits que se necesitan para representar los coeficientes de filtro, se pueden fusionar diferentes clases. Sin embargo, a diferencia de JCTVC-E603, se puede fusionar cualquier conjunto de clases, incluso las clases que tienen valores no consecutivos de C . La información con respecto a qué clases se fusionan se proporciona al enviar para cada una de las 25 clases un índice i_C . Las

clases que tienen el mismo índice i_c comparten los mismos coeficientes de filtro que se codifican. El índice i_c se codifica con un método de longitud fija truncada.

[0099] De forma similar, también se puede usar la etiqueta `forceCoeff0`. Si la etiqueta `forceCoeff0` es igual a 1, una etiqueta de un bit, indicada como *CodedVarBin*, se señala de forma adicional para cada uno de los grupos fusionados (todos los filtros a ser codificados) para indicar si los coeficientes de filtro señalizados son todos cero. Además, si `forceCoeff0` es igual a 1, se inhabilita la codificación predictiva (es decir, codificar la diferencia entre el filtro actual y el filtro codificado previamente). Cuando se permite la predicción de filtros fijos, los filtros que deben ser señalizados/codificados que se mencionaron anteriormente son las diferencias entre el filtro aplicado a la imagen reconstruida y el filtro fijo seleccionado. Otra información, tal como coeficientes, se codifica de la misma forma que en JEM2.0.

[0100] Puesto que GALF es una forma de ALF, la presente divulgación puede usar el término “ALF” para referirse a ALF y GALF.

[0101] Los diseños actuales para predicción temporal de los filtros en ALF y GALF tienen varios inconvenientes. Por ejemplo, si una imagen usa la codificación explícita de filtros, luego de decodificar la imagen, se pueden agregar los filtros de ALF correspondientes a una matriz de filtros de ALF para predicción temporal, independientemente de las capas temporales. Es decir, luego de decodificar la imagen, un codificador de video puede incluir un conjunto de parámetros de ALF en una entrada en la matriz. El conjunto de parámetros de ALF puede incluir coeficientes de filtro e información de fusión de grupos para cada uno de los filtros de ALF usados en la imagen. Este diseño provoca una falla cuando se decodifica un subconjunto de capas temporales en determinadas configuraciones, tales como el acceso aleatorio. Un ejemplo se proporciona en la Figura 3 en donde el tamaño de GOP es igual a 16. En el ejemplo de la Figura 3, son compatibles cinco capas temporales (indicadas mediante T_0 a T_4). El orden de codificación/decodificación de las imágenes es: contador de orden de imágenes (POC) 0 [T_0], POC 16 [T_0], POC8 [T_1], POC4 [T_2], POC2 [T_3], POC1 [T_4], POC3 [T_4], POC6 [T_3], POC5 [T_4], POC7 [T_4], POC12 [T_2], POC10 [T_3], POC9 [T_4], POC11 [T_4], POC14 [T_3], POC13 [T_4], POC15 [T_4]. Las flechas con diferentes patrones de línea señalan imágenes que pueden usar las imágenes señaladas como imágenes de referencia. Cabe destacar que la Figura 3 omite ciertas flechas por razones de claridad.

[0102] La Figura 4A ilustra una matriz 50 para almacenar parámetros de filtro. La Figura 4B ilustra un estado diferente de matriz 50. Suponiendo que cada imagen se

codifica con ALF habilitado y los filtros de ALF para cada imagen se señalizan de forma explícita, antes de decodificar POC3 de la Figura 3, la matriz para filtros almacenados tiene el estado que se muestra en la Figura 4A. Después de decodificar POC3 y antes de decodificar POC6 de la Figura 3, la matriz para filtros de ALF almacenados se actualiza como se muestra en la Figura 4B. Como se muestra en el ejemplo de la Figura 4B, los filtros para POC0 se reemplazaron con los filtros para POC3 ya que los filtros se reemplazan de acuerdo con el esquema de FIFO y los filtros para POC0 fueron los primeros que se agregaron a la matriz 50.

[0103] Por lo tanto, para decodificar POC6 con el índice de capa temporal (Templdx) igual a 3, los filtros de POC1, POC3 con el índice de capa temporal igual a 4 necesitan ser decodificados. Esto contradice el espíritu de escalabilidad temporal, en donde la decodificación de una imagen con un cierto valor de Templdx no debe depender de imágenes con un valor de Templdx superior.

[0104] Un segundo inconveniente de los diseños actuales para la predicción temporal de los filtros en ALF es que, cuando la predicción temporal de los filtros de ALF se habilita para un segmento, en algunos ejemplos, todos los filtros de ALF de un cierto cuadro codificado previamente deben ser heredados. Esto significa que la fusión de clases y coeficientes de filtro se reutiliza de forma directa sin la posibilidad de modificar levemente las clases y coeficientes de filtro para capturar de una mejor forma las características para un segmento actual.

[0105] Las siguientes técnicas se proponen para resolver uno o más de los inconvenientes de los diseños actuales para la predicción temporal de los filtros en ALF mencionados anteriormente. Las técnicas que se enumeran a continuación se pueden aplicar de forma individual. De forma alternativa, se puede aplicar cualquier combinación de estas.

[0106] De acuerdo con una primera técnica, se pueden asignar múltiples matrices para almacenar uno o más conjuntos de filtros de ALF codificados previamente. En otras palabras, un codificador de video puede almacenar conjuntos de parámetros de ALF en una pluralidad de matrices. Cada matriz corresponde a un índice de capa temporal asignado (Templdx, que es igual a TemporalId definido en la especificación de HEVC). De acuerdo con una primera técnica, cada matriz contiene únicamente los parámetros de ALF de imágenes con el mismo Templdx o Templdx inferior. Un segmento (u otra unidad para llevar a cabo el ALF) con el Templdx puede seleccionar un conjunto de filtros comprendido en esta matriz. En otras palabras, un codificador de video puede aplicar, a las muestras en un bloque del segmento, un filtro de ALF con

base en parámetros de ALF en la matriz que corresponde al TempIdx del segmento. Para una región que se codifica con ALF habilitado y, suponiendo que los parámetros de ALF se señalizan de forma explícita (es decir, sin predicción temporal), el conjunto de parámetros de ALF para esta región se puede agregar a la matriz asociada al mismo TempIdx o uno superior. Esto podría resolver los inconvenientes descritos anteriormente con respecto a la matriz de los parámetros de ALF almacenados, que incluyen uno o más parámetros de ALF que corresponden a los filtros de ALF usados en imágenes de capas temporales superiores a la capa temporal de la imagen actual.

[0107] La Figura 5 ilustra una pluralidad de matrices 60A-60E (en conjunto denominadas “matrices 60”) que corresponden a diferentes capas temporales, de acuerdo con una técnica de la presente divulgación. En el ejemplo de la Figura 5, suponiendo que cada imagen de la Figura 3 está codificada con ALF habilitado y los filtros de ALF para cada imagen son señalados explícitamente, antes de decodificar POC6 de la Figura 3, las matrices para los filtros de ALF almacenados tienen los estados mostrados en la Figura 5.

[0108] En el ejemplo de la Figura 5, puesto que POC6 se encuentra en la capa temporal T_3 , un codificador de video puede usar los filtros de ALF de la matriz 60D. Por lo tanto, a diferencia del ejemplo de la Figura 4B, si POC1 se decodifica o no, no repercute en qué filtros de ALF están disponibles para usar al decodificar POC 6.

[0109] De este modo, de acuerdo con la primera técnica, el codificador de video 20 puede generar una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video. Una región actual (por ejemplo, un segmento u otro tipo de unidad para realizar ALF) de la imagen actual se asocia a un índice temporal (es decir, un índice de capa temporal) que indica una capa temporal a la que pertenece la región actual. Además, el codificador de video 20 reconstruye la totalidad o parte de la imagen actual. El codificador de video 20 almacena, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video decodificados antes que la región actual. Por ejemplo, para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, el codificador de video 20 puede almacenar, en la matriz respectiva, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video que son decodificados antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva. Cada matriz respectiva de la pluralidad de matrices

corresponde a una capa temporal distinta respectiva. Además, el codificador de video 20 determina, en función de un conjunto seleccionado de parámetros de ALF en una de las matrices correspondientes a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En algunos ejemplos, el codificador de video 20 puede determinar el conjunto seleccionado de parámetros de ALF en función de un análisis de tasa-distorsión de los conjuntos de parámetros de ALF en las matrices. El codificador de video 20 puede señalar un índice del conjunto seleccionado de parámetros de ALF en la corriente de bits. Además, en este ejemplo, el codificador de video 20 aplica, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual. La aplicación del filtrado en bucle adaptativo a la región actual puede comprender la aplicación de un filtro de ALF a uno o más, pero no necesariamente a todos, de los bloques en la región actual. Luego de aplicar el filtrado en bucle adaptativo a la región actual, el codificador de video 20 puede usar la región actual para predicción de una imagen posterior de los datos de video.

[0110] De forma similar, de acuerdo con la primera técnica, el decodificador de video 30 puede recibir una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video. Una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual. El decodificador de video 30 puede luego reconstruir la totalidad o parte de la imagen actual. Además, el decodificador de video 30 almacena, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video decodificados antes que la imagen actual. Cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal distinta respectiva. Por ejemplo, para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, el decodificador de video 30 puede almacenar, en la matriz respectiva, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video que son decodificados antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva. El decodificador de video 30 determina, en función de un conjunto seleccionado de parámetros de ALF en la matriz correspondiente a la capa temporal a la que pertenece la región actual o matrices de la pluralidad de matrices que corresponde a una capa

temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En algunos ejemplos, el decodificador de video 30 determina el conjunto seleccionado de parámetros de ALF en función de un índice señalizado en la corriente de bits. El decodificador de video 30 puede luego aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual. La aplicación del filtro de ALF a la región actual puede comprender la aplicación del filtro de ALF a uno o más, pero no necesariamente a todos, de los bloques en la región actual.

[0111] Cada matriz asignada mediante $Templdx$ puede comprender conjuntos de filtros decodificados previamente asociados a imágenes con un $Templdx$ igual o inferior. Por ejemplo, la matriz k se asigna para ser asociada a $Templdx$ igual a k , y solo contiene los conjuntos completos o los subconjuntos de filtros (por ejemplo, parámetros de ALF para filtros) de imágenes con $Templdx$ igual o inferior a k .

[0112] Por ende, para cada matriz respectiva de la pluralidad de matrices, un codificador de video (por ejemplo, el codificador de video 20 o el decodificador de video 30) puede almacenar, en la matriz respectiva, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de las regiones de las imágenes de los datos de video decodificados antes que la imagen actual perteneciente a la capa temporal que corresponde a la matriz respectiva y perteneciente a capas temporales inferiores a la capa temporal que corresponde a la matriz respectiva.

[0113] En algunos ejemplos, las cantidades de conjuntos de filtros asociados a distintas matrices pueden ser distintas (que podría ser mayor de o igual a 0). De manera alternativa, en algunos ejemplos, las cantidades de los conjuntos de filtros asociados a distintas capas temporales pueden ser distintas y pueden depender del índice de capa temporal. Por ende, en algunos ejemplos, al menos dos de la pluralidad de matrices incluyen distintas cantidades de conjuntos de parámetros de ALF. Por ejemplo, en el ejemplo de la Figura 5, puede no ser necesario tener cinco localizaciones en la matriz 60A porque, en un GOP de 16 imágenes, nunca habrá más de dos imágenes en capa temporal T_0 . Por ende, la matriz 60A solo puede tener dos localizaciones. De manera similar, en el ejemplo de la Figura 5, en un GOP de 16 imágenes, habrá como máximo una imagen en capa temporal T_1 . Por ende, la matriz 60B solo puede tener tres localizaciones.

[0114] En algunos ejemplos, luego de codificar determinado segmento/unidad para llevar a cabo ALF, un codificador de video puede usar el conjunto de filtros asociados al segmento para actualizar aquellas matrices asociadas a un $Templdx$ igual o

superior. Por ejemplo, un codificador de video puede almacenar, en la matriz correspondiente a la capa temporal a la que pertenece la región actual (y, en algunos casos, matrices que corresponden a capas temporales que corresponden a capas temporales superiores a la capa temporal a la que pertenece la región actual), un conjunto de parámetros de ALF aplicable a un región actual (es decir, segmento u otra unidad para llevar a cabo ALF). Por ejemplo, en el ejemplo de la Figura 3 y la Figura 5, si la región actual está en una imagen asociada a POC 8, el codificador de video puede actualizar las matrices 60B, 60C, 60D, y 60E para incluir el conjunto de parámetros de ALF aplicable a la región actual.

[0115] En algunos ejemplos, el valor POC asociado a cada conjunto de filtros (por ejemplo, un conjunto de parámetros de ALF) también se puede grabar. Por ende, un codificador de video puede almacenar, en la matriz correspondiente a la capa temporal a la que pertenece una región actual de una imagen actual, un valor POC de la imagen actual. En un ejemplo, cuando se selecciona un filtro como un candidato de una matriz determinada para predicción temporal de ALF, se puede requerir que el valor POC asociado al filtro sea igual a un valor POC de una de las imágenes de referencia en listas de imágenes de referencia actuales. Por ejemplo, además de almacenar los parámetros de ALF para filtros de ALF usados por la imagen en POC0 en la Figura 5, un codificador de video puede almacenar en la matriz 60A datos que indican un valor de POC0. En este ejemplo, si la imagen en POC0 no es en una imagen de referencia de la imagen en POC6 cuando se codifica una región de la imagen en POC6, no se permite que el codificador de video 20 seleccione un filtro de ALF de entre los filtros de ALF almacenados en la matriz 60A para la imagen en POC0.

[0116] De acuerdo con una segunda técnica, una matriz todavía se usa para almacenar conjuntos de filtros de ALF codificados previamente. Además de filtros, para cada conjunto (que puede contener múltiples filtros usados para codificar un segmento/imagen), el índice de capa temporal (TempIdx) asociado a los conjuntos de filtros también se graba. En otras palabras, los índices de capa temporal se pueden almacenar junto con los parámetros de ALF para filtros de ALF.

[0117] En algunos ejemplos con base en la segunda técnica, el tamaño de la matriz se puede ajustar a (cantidad posible de capas temporales)*(cantidad máxima de conjuntos de filtros para predicción temporal para un segmento/imagen u otra unidad para uso de ALF). En un ejemplo, la cantidad de posibles capas temporales puede depender de una estructura de codificación (por ejemplo, cuántos niveles son compatibles en la estructura B de jerarquía) o una etiqueta de verificación de bajo

retardo (NoBackwardPredFlag en la especificación de HEVC).

[0118] En un ejemplo, una cantidad máxima de conjuntos de filtros para predicción temporal para un segmento/imagen u otra unidad para uso de ALF puede predefinirse o señalizarse o depender del TempIdx. En un ejemplo, la cantidad de posibles capas temporales se ajusta a 5 y la cantidad máxima de conjuntos de filtros para predicción temporal para un segmento/imagen u otra unidad para uso de ALF se ajusta a 6. Cuando se codifica un segmento/imagen, los posibles candidatos para predicción temporal se pueden definir revisando los conjuntos incluidos en la matriz y la totalidad o parte de los conjuntos de filtros con un TempIdx igual o inferior se tratan como candidatos eficaces.

[0119] Luego de codificar determinado segmento/unidad para llevar a cabo ALF, el conjunto de filtros asociado al segmento y el TempIdx asociado se pueden usar para actualizar la matriz. Por ejemplo, un codificador de video (por ejemplo, el codificador de video 20 o el decodificador de video 30) puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz, un conjunto de parámetros de ALF aplicable para una región. En este ejemplo, el codificador o decodificador puede almacenar, en la matriz, el conjunto de parámetros de ALF aplicable. El codificador o decodificador también puede almacenar el conjunto de parámetros de ALF aplicable en una o más de las matrices que corresponden a capas temporales superiores a la capa temporal a la que pertenece la región actual. En este ejemplo, el codificador de video puede no almacenar parámetros de ALF en la matriz si los parámetros de ALF no se señalaron explícitamente en la corriente de bits. En algunos ejemplos, el codificador o decodificador solo almacena el conjunto de parámetros de ALF aplicable en la matriz si el conjunto de parámetros de ALF aplicable aún no se ha almacenado en la matriz.

[0120] La Figura 6 ilustra una matriz 70 para almacenar parámetros de ALF y valores del índice de capa temporal (TempIdx), de acuerdo con la segunda técnica de la presente divulgación. En el ejemplo de la Figura 6, la cantidad de posibles capas temporales es 5 y la cantidad máxima de conjuntos de filtros para predicción temporal para una región se ajusta a 6, lo que da como resultado la matriz 70 que contiene 30 entradas. En el ejemplo de la Figura 6, suponiendo que cada imagen de la Figura 3 está codificada con ALF habilitado y los filtros de ALF para cada imagen son señalizados explícitamente, antes que decodificar POC6 de la Figura 3, las matrices para los filtros de ALF almacenados tienen los estados mostrados en la Figura 6.

[0121] En el ejemplo de la Figura 6, un codificador de video puede revisar los valores

de TempIdx almacenados en la matriz 70 para determinar cuáles de las entradas de la matriz 70 almacenan parámetros de ALF que el codificador de video puede usar como predictores de parámetros de ALF usados para codificar POC6. Al hacer eso, el codificador de video puede ignorar cualquier entrada que especifique valores de TempIdx superiores a T_3 (es decir, el TempIdx para POC6). A diferencia del ejemplo de la Figura 4B, los filtros para POC0 no se sobrescriben por los filtros para POC3.

[0122] De este modo, de acuerdo con la segunda técnica de la presente divulgación, el codificador de video 20 puede generar una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video. Una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual. Además, el codificador de video 20 puede reconstruir la imagen actual. El codificador de video 20 también almacena, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. Además, el codificador de video 20 almacena, en la matriz, índices de capa temporal asociados a los conjuntos de parámetros de ALF. Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. En este ejemplo, el codificador de video 20 determina, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual o una capa temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Luego, el codificador de video 20 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual. Luego de aplicar el filtrado en bucle adaptativo a la región actual, el codificador de video 20 puede usar la región actual para predicción de una imagen posterior de los datos de video.

[0123] De forma similar, de acuerdo con la segunda técnica de la presente divulgación, el decodificador de video 30 puede recibir una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video. Una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual. Además, el decodificador de video 30 puede reconstruir la imagen actual. En este ejemplo, el decodificador de video 30 almacena, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados con

anterioridad a la imagen actual. Además, el decodificador de video 30 almacena, en la matriz, índices de capa temporal asociados a los conjuntos de parámetros de ALF. Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. En este ejemplo, el decodificador de video 30 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Además, en este ejemplo, el decodificador de video 30 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual.

[0124] En algunos ejemplos con base en la segunda técnica de la presente divulgación, el valor POC asociado a cada conjunto de filtros de ALF también se puede grabar. Por ejemplo, un codificador de video también puede almacenar, en una matriz (por ejemplo, matriz 70), un valor POC de una imagen actual de la que se codifican parámetros de ALF de manera explícita. Por lo tanto, en este ejemplo, luego de codificar/decodificar una pluralidad de múltiples imágenes, el codificador de video almacenó, en la matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes de una imagen actual nueva.

[0125] En un ejemplo, los candidatos posibles para predicción temporal se pueden definir revisando los conjuntos incluidos en la matriz, todos o algunos de los conjuntos de filtros con igual o menor $TempIdx$ y aquellos con un valor POC igual a un valor POC de uno de la imagen de referencia en una lista de imágenes de referencia actual se pueden tratar como candidatos eficaces. Por ejemplo, un codificador de video puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece una región actual de una imagen actual o una capa temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En este ejemplo, se requiere que, cuando se determina el conjunto de parámetros de ALF aplicable para la región actual, un valor POC asociado al conjunto de parámetros de ALF aplicable para la región actual sea igual a un valor POC de una imagen de referencia en una lista de imágenes de referencia de la imagen actual.

[0126] En algunos ejemplos con base en la segunda técnica de la presente divulgación, cada filtro de ALF que se debe almacenar en la matriz será asociado a

una imagen de referencia que se incluye en un conjunto de imágenes de referencia de la imagen actual (estas imágenes también estarán disponibles en la memoria de imágenes decodificadas). Es decir, si una imagen no se incluye en el conjunto de imágenes de referencia de la imagen actual, los filtros asociados a la imagen actual no se pueden almacenar y usar para predicción temporal de ALF.

[0127] En algunos ejemplos con base en la segunda técnica de la presente divulgación, el tamaño de la matriz puede depender del tamaño de un conjunto de imágenes de referencia. Por ejemplo, el tamaño de la matriz puede ser igual a una cantidad máxima de imágenes de referencia que se permite tener en un conjunto de imágenes de referencia.

[0128] En algunos ejemplos, un codificador de video no genera una lista específicamente para parámetros de filtros de ALF (es decir, parámetros de ALF), pero la lista es la misma que las listas de imágenes de referencia que se generan para el segmento actual. En este caso, los parámetros de ALF asociados a imágenes de referencia de la región actual se almacenan directamente junto con otra información (tal como muestras de reconstrucción, información de movimiento de cada bloque con una región) requerida por el almacenamiento de imágenes de referencia. Como otra alternativa, la lista de parámetros de filtros de ALF se establece igual al conjunto de imágenes de referencia del segmento (o imagen) actual.

[0129] En otro ejemplo en donde cada filtro de ALF almacenado en la matriz (por ejemplo, matriz 70) se asocia a una imagen de referencia incluida en el conjunto de imágenes de referencia de la imagen actual, la lista (matriz) de parámetros de filtros de ALF (asociada a imágenes de referencia incluidas en el conjunto de imágenes de referencia de la imagen actual) se genera por separado independientemente de las listas de imágenes de referencia para el segmento actual. Para la generación eficaz de una lista eficaz de parámetros de filtros de ALF, tal que los conjuntos de parámetros de filtros de ALF usados más frecuentemente estén en las primeras posiciones en la lista de parámetros de filtros de ALF, los elementos sintácticos para la señalización de un orden particular de los conjuntos candidatos de parámetros de filtros de ALF en la lista de parámetros de filtros de ALF se pueden incluir en un encabezado del segmento, de forma similar a la sintaxis de modificación de la lista de imágenes de referencia en el encabezado del segmento.

[0130] De acuerdo con una tercera técnica, en vez de usar la regla FIFO para actualizar unas matrices para filtros de ALF almacenados, se propone considerar, además, las diferencias del contador de orden de imágenes (POC) para actualizar las

matrices. Por ejemplo, si una matriz (por ejemplo, matriz 50 de la Figura 4A y la Figura 4B, una de las matrices 60 de la Figura 5, o matriz 70 de la Figura 6) está completa, un codificador de video puede determinar qué entrada en la matriz almacena filtros de ALF asociados al valor POC más distinto de un valor POC de una imagen actual. En un ejemplo con base en la primera técnica, cuando un conjunto de parámetros de ALF se señala de manera explícita para una región de una imagen actual, un codificador de video puede determinar, en función de las diferencias entre un valor POC de la imagen actual y valores POC asociados a conjuntos de parámetros de ALF, qué conjunto de parámetros de ALF en la matriz que corresponde a la capa temporal a la que pertenece la región actual reemplazar con el conjunto de parámetros de ALF aplicable para la región actual. En un ejemplo con base en la segunda técnica, cuando un conjunto de parámetros de ALF se señala de manera explícita para una región de una imagen actual, un codificador de video puede determinar, en función de las diferencias entre un valor POC de la imagen actual y valores POC asociados a conjuntos de parámetros de ALF, qué conjunto de parámetros de ALF en la matriz reemplazar con el conjunto de parámetros de ALF aplicable para la región actual.

[0131] En algunos ejemplos, se puede definir una lista separada para seleccionar filtros de un conjunto de imágenes de referencia, que es distinto de seleccionar imágenes de referencia de un conjunto de imágenes de referencia. En este caso, los filtros seleccionados pueden ser de una imagen que no se incluye en ninguna lista de imágenes de referencia de segmento/mosaico/imagen actual.

[0132] De acuerdo con una cuarta técnica, la señalización de un índice de un conjunto/subconjunto seleccionado de filtros para la predicción temporal de ALF puede depender de un índice de capa temporal. Un subconjunto de filtros para predicción temporal de ALF es un conjunto parcial de filtros de ALF. Por ejemplo, actualmente, podría haber 25 filtros de ALF por imagen. En este ejemplo, cuando se usa predicción temporal, el codificador de video puede elegir 10 en vez de 25 filtros de ALF para ser asociados a una imagen. En un ejemplo, el método de binarización unaria truncada se puede usar para codificar el índice seleccionado de un conjunto de filtros y el valor máximo de la cantidad permitida de conjuntos depende del índice de capa temporal.

[0133] Por ejemplo, de acuerdo con un ejemplo de la cuarta técnica, el codificador de video 20 puede incluir, en una corriente de bits, un elemento sintáctico que indica un índice de un conjunto de parámetros de ALF seleccionado. De forma similar, el decodificador de video 30 puede obtener, de la corriente de bits, un elemento sintáctico que indica un índice de un conjunto de parámetros de ALF seleccionado. El

conjunto de parámetros de ALF seleccionado puede estar en una de las matrices de un tipo usadas en la primera técnica o la matriz de un tipo usada en la segunda técnica. En este ejemplo, el codificador de video 20 y/o decodificador de video 30 pueden determinar, en función del conjunto seleccionado de parámetros de ALF en la matriz, un conjunto de parámetros de ALF aplicable para la región actual. El codificador de video 20 y/o decodificador de video 30 pueden aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, un filtro de ALF a la región actual. En este ejemplo, un formato del elemento sintáctico depende de un índice de capa temporal. Por ejemplo, un método de binarización unaria truncada se puede usar para codificar el elemento sintáctico y un valor máximo de la cantidad permitida de conjuntos de parámetros de ALF depende del índice de capa temporal.

[0134] En algunos ejemplos con base en la cuarta técnica, la señalización del índice también puede depender de las diferencias del POC. En otras palabras, en el contexto del ejemplo del párrafo anterior, el formato del elemento sintáctico también depende de diferencias del POC. Por ejemplo, si el índice es 0, el conjunto de parámetros de ALF seleccionado se asocia a la imagen con el valor POC más cercano a un valor POC de una imagen actual; si el índice es 1, el conjunto de parámetros de ALF seleccionado se asocia a la imagen con un siguiente valor POC más cercano al valor POC de la imagen actual, y así sucesivamente. En este ejemplo, si dos o más de los conjuntos de parámetros de ALF en la matriz o matrices se asocian a imágenes que tienen la misma distancia de POC de la imagen actual, los conjuntos de parámetros de ALF asociados a imágenes con valores POC inferiores (o, en otros ejemplos, superiores) se asocian a valores de índice inferiores.

[0135] De acuerdo con una quinta técnica, en vez de heredar coeficientes de filtro e información de fusión de clases, se propone que solo la información de fusión de clases se pueda heredar. Es decir, los índices de filtros para distintas clases podrían heredarse de información previamente codificada. Alternativamente, además, se les puede asignar a matrices separadas una matriz para grabar los índices de filtros para cada clase y la otra para grabar los coeficientes de filtro.

[0136] Por lo tanto, en un ejemplo de acuerdo con la quinta técnica, un codificador de video puede almacenar, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual, cada matriz respectiva de la pluralidad de matrices que corresponden a una capa temporal distinta respectiva. En este ejemplo, el codificador de video puede determinar, del conjunto de parámetros de ALF

en la matriz que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, la información de fusión de clases y no coeficientes de filtro.

[0137] Además, en algunos ejemplos, el codificador de video puede almacenar, en una segunda pluralidad de matrices, conjuntos de coeficientes de filtro usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que una imagen actual, en donde cada matriz respectiva de la segunda pluralidad de matrices corresponde a una capa temporal distinta respectiva. Como parte de la determinación del conjunto de parámetros de ALF aplicable para la región actual, el codificador de video puede determinar, en función de un conjunto de coeficientes de filtro en una de las matrices de la segunda pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual y en función del conjunto de parámetros de ALF en una matriz en la primera pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, el conjunto de parámetros de ALF aplicable.

[0138] En un ejemplo de acuerdo con la quinta técnica, un codificador de video puede almacenar, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. En este ejemplo, el codificador de video puede determinar, del conjunto de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual o una capa temporal inferior a la capa temporal a la que pertenece una región actual, información de fusión de clases y no coeficientes de filtro.

[0139] Además, en algunos ejemplos, el codificador de video puede almacenar, en una segunda matriz, conjuntos de coeficientes de filtro usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. En tales ejemplos, el codificador de video puede almacenar, en la segunda matriz, índices de capa temporal asociados a los conjuntos de coeficientes de filtro. Un índice de capa temporal asociado a un conjunto de coeficientes de filtro indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. Como parte de la determinación del conjunto de parámetros de ALF aplicable para la región actual, el codificador de video puede determinar, en función de un conjunto de coeficientes de filtro en una de las matrices

de la segunda pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual y en función del conjunto de parámetros de ALF en una matriz en la primera pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, el conjunto de parámetros de ALF aplicable para la región actual.

[0140] De acuerdo con una sexta técnica, en vez de heredar coeficientes de filtro e información de fusión de clases, se propone que solo los coeficientes de filtro se puedan heredar. Es decir, para el segmento/imagen actual, la relación entre índice de clase e índice de filtro se puede señalar adicionalmente, incluso se usa predicción temporal.

[0141] Por lo tanto, de acuerdo con un ejemplo de la sexta técnica, un codificador de video puede almacenar, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. Cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal distinta respectiva. En este ejemplo, como parte de la determinación de un conjunto de parámetros de ALF aplicable para la región actual, el codificador de video puede determinar, del conjunto de parámetros de ALF en la matriz que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, coeficientes de filtro y no información de fusión de clases.

[0142] Además, en algunos ejemplos, el codificador de video puede almacenar, en una segunda pluralidad de matrices, conjuntos de información de fusión de clases usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. Cada matriz respectiva de la segunda pluralidad de matrices corresponde a una capa temporal distinta respectiva. Como parte de la determinación del conjunto de parámetros de ALF aplicable para la región actual, el codificador de video puede determinar, en función de un conjunto de información de fusión de clases en una de las matrices de la segunda pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual y en función del conjunto de parámetros de ALF en una matriz en la primera pluralidad de matrices que corresponde a la capa temporal a la que pertenece

la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, el conjunto de parámetros de ALF aplicable.

[0143] De acuerdo con otro ejemplo de la sexta técnica, un codificador de video puede almacenar, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. En este ejemplo, el codificador de video puede determinar, del conjunto de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual o una capa temporal inferior a la capa temporal a la que pertenece la región actual, coeficientes de filtro y no información de fusión de clases.

[0144] Además, en algunos ejemplos, el codificador de video puede almacenar, en una segunda matriz, conjuntos de información de fusión de clases usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. En este ejemplo, el codificador de video también puede almacenar, en la segunda matriz, índices de capa temporal asociados a los conjuntos de información de fusión de clases. Un índice de capa temporal asociado a un conjunto de coeficientes de filtro indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. Como parte de la determinación del conjunto de parámetros de ALF aplicable para la región actual, el codificador de video puede determinar, en función de un conjunto de coeficientes de filtro en una de las matrices de la segunda pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual y en función del conjunto de parámetros de ALF en una matriz en la primera pluralidad de matrices que corresponde a la capa temporal a la que pertenece la región actual o que corresponde a una capa temporal inferior a la capa temporal a la que pertenece la región actual, el conjunto de parámetros de ALF aplicable para la región actual.

[0145] De acuerdo con una séptima técnica, aun cuando se usa la predicción temporal, las diferencias entre los filtros almacenados seleccionados y los filtros derivados se pueden señalar adicionalmente. En un ejemplo, aún se puede usar el diseño actual para la señalización de la predicción temporal habilitando etiqueta e índice de un conjunto de filtros. Además, una etiqueta se puede usar para indicar si señalar las diferencias de filtros o no. En caso de que sí, las diferencias se pueden señalar adicionalmente. En algunos ejemplos, los filtros de cuadros o segmentos previamente codificados se pueden agregar y tratar como parte de filtros fijos. En este

caso, el tamaño de los filtros fijos y coeficientes de filtros fijos se puede cambiar con adaptación. Alternativamente, además, en algunos ejemplos, cuando un filtro de un conjunto se agrega a los filtros fijos, se aplicará reducción para evitar duplicaciones.

[0146] En un ejemplo de acuerdo con la séptima técnica, el codificador de video 20 puede determinar, en función de un conjunto de parámetros de ALF seleccionado en una matriz que corresponde a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Alternativamente, en este ejemplo, el codificador de video 20 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En cualquiera de estos casos, el codificador de video 20 puede incluir, en la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual. En algunos ejemplos, el codificador de video 20 puede incluir, en la corriente de bits, un elemento sintáctico que indica si la corriente de bits incluye la indicación de la diferencia.

[0147] En otro ejemplo de acuerdo con la séptima técnica, el decodificador de video 30 puede determinar, en función de un conjunto de parámetros de ALF seleccionado en una matriz que corresponde a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Alternativamente, en este ejemplo, el decodificador de video 30 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En cualquiera de estos casos, el decodificador de video 30 puede obtener, de la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual. En este ejemplo, como parte de la determinación del conjunto de parámetros de ALF aplicable para la región actual, el decodificador de video 30 puede determinar, en función del conjunto de parámetros de ALF seleccionado y la diferencia, el conjunto de parámetros de ALF aplicable para la región actual. En algunos ejemplos, el decodificador de video 30 puede obtener, de la corriente de bits, un elemento sintáctico que indica si la corriente de bits incluye la indicación de la diferencia.

[0148] De acuerdo con una octava técnica, uno o más conjuntos de filtros de ALF pueden ser almacenados en conjuntos de parámetros (por ejemplo, conjuntos de

parámetros de secuencias o conjuntos de parámetros de imágenes) para que las imágenes incluso en distintas secuencias de video codificadas puedan usarlos. Para evitar problemas de resiliencia de errores o problemas de acceso aleatorio, no se permite actualizar los conjuntos de filtros de ALF en conjuntos de parámetros usando filtros de ALF señalizados en encabezados de segmento. Por ejemplo, cuando se codifica una corriente de bits, un codificador de video puede almacenar, en una matriz, conjuntos de filtros de ALF especificados en un conjunto de parámetros de la corriente de bits. En este ejemplo, los encabezados de segmento pueden incluir parámetros de ALF que definen filtros de ALF adicionales o diferencias de filtros. Un encabezado de segmento es una parte de un segmento codificado que contiene los elementos de datos que pertenecen a la primera o a todas las unidades de decodificación en árbol representadas en el segmento.

[0149] La Figura 7 es un diagrama de bloques que ilustra un codificador de video 20 de ejemplo que puede implementar las técnicas de la presente divulgación. La Figura 7 se proporciona con fines explicativos y no se debe considerar limitante de las técnicas ampliamente ejemplificadas y descritas en la presente divulgación. Las técnicas de la presente divulgación pueden ser aplicables a diversos métodos o estándares de codificación.

[0150] El circuito de procesamiento incluye el codificador de video 20, en donde el codificador de video 20 se configura para realizar una o más de las técnicas de ejemplo descritas en la presente divulgación. Por ejemplo, el codificador de video 20 incluye un circuito integrado, y las diversas unidades ilustradas en la Figura 5 se pueden formar como bloques de circuitos de *hardware* que están interconectados con un bus de circuito. Estos bloques de circuitos de *hardware* pueden ser bloques de circuitos separados o dos o más de las unidades se pueden combinar en un bloque de circuitos de *hardware* común. Los bloques de circuitos de *hardware* se pueden formar como combinación de componentes eléctricos que forman bloques de operación tales como unidades lógicas aritméticas (ALU), unidades de función elemental (EFU), así como bloques lógicos tales como AND, OR, NAND, NOR, XOR, XNOR y otros bloques lógicos similares.

[0151] En algunos ejemplos, una o más de las unidades ilustradas en la Figura 7 pueden ser unidades de *software* que se ejecutan en el circuito de procesamiento. En tales ejemplos, el código de objeto para estas unidades de *software* se almacena en la memoria. Un sistema operativo puede ocasionar que el codificador de video 20 recupere el código de objeto y ejecute el código de objeto, que hace que el codificador

de video 20 realice operaciones para implementar las técnicas de ejemplo. En algunos ejemplos, las unidades de *software* pueden ser *firmware* que el codificador de video 20 ejecuta al encenderse. En consecuencia, el codificador de video 20 es un componente estructural que tiene *hardware* que realiza las técnicas de ejemplo o tiene *software/firmware* que se ejecuta en el *hardware* para especializar el hardware para realizar las técnicas de ejemplo.

[0152] En el ejemplo de la Figura 7, el codificador de video 20 incluye una unidad de procesamiento de predicción 100, memoria de datos de video 101, una unidad de generación residual 102, una unidad de procesamiento de la transformada 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de procesamiento de la transformada inversa 110, una unidad de reconstrucción 112, una unidad de filtro 114, una memoria de imagen decodificada 116, y una unidad de codificación entrópica 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de interpredicción 120 y una unidad de procesamiento de intrapredicción 126. La unidad de procesamiento de interpredicción 120 puede incluir una unidad de estimación de movimiento y una unidad de compensación de movimiento (no se muestra).

[0153] La memoria de datos de video 101 se puede configurar para almacenar los datos de video que desean ser codificados por los componentes del codificador de video 20. Los datos de video almacenados en la memoria de datos de video 101 se pueden obtener, por ejemplo, de la fuente de video 18. La memoria de imagen decodificada 116 puede ser una memoria de imagen de referencia que almacena datos de video de referencia para usar en la codificación de datos de video mediante el codificador de video 20, por ejemplo, en modos de intra- o intercodificación. La memoria de datos de video 101 y la memoria de imagen decodificada 116 se pueden formar con una variedad de dispositivos de memoria, tal como memoria dinámica de acceso aleatorio (DRAM), que incluye DRAM síncrona (SDRAM), RAM magnetoresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de video 101 y la memoria de imagen decodificada 116 se pueden proporcionar mediante el mismo dispositivo de memoria o mediante dispositivos de memoria diferentes. En diversos ejemplos, la memoria de datos de video 101 puede ser interna (*on-chip*) con otros componentes del codificador de video 20 o externa (*off-chip*) con respecto a aquellos componentes. La memoria de datos de video 101 puede ser la misma que o parte de los medios de almacenamiento 19 de la Figura 1.

[0154] El codificador de video 20 recibe datos de video. El codificador de video 20 puede codificar cada CTU en un segmento de una imagen de los datos de video. Cada una de las CTU se puede asociar a bloques de codificación en árbol (CTB) de luminancia del mismo tamaño y CTB correspondientes de la imagen. Como parte de la codificación de una CTU, la unidad de procesamiento de predicción 100 puede realizar particiones para dividir los CTB de las CTU en bloques progresivamente más pequeños. Los bloques más pequeños pueden ser bloques de codificación de las CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede particionar un CTB asociado a una CTU de acuerdo con una estructura en árbol.

[0155] El codificador de video 20 puede codificar las CU de una CTU para generar representaciones codificadas de las CU (es decir, CU codificadas). Como parte de la codificación de una CU, la unidad de procesamiento de predicción 100 puede particionar los bloques de codificación asociados a la CU entre una o más PU de la CU. Por lo tanto, cada PU se puede asociar a un bloque de predicción de luminancia y los correspondientes bloques de predicción de crominancia. El codificador de video 20 y decodificador de video 30 pueden ser compatibles con las PU que tienen diversos tamaños. Como se indicó anteriormente, 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 un bloque de predicción de luminancia de la PU. Suponiendo que el tamaño de una CU particular es $2N \times 2N$, el codificador de video 20 y decodificador de video 30 pueden ser compatibles con tamaños de PU de $2N \times 2N$ o $N \times N$ para intrapredicción, y tamaños simétricos de PU de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$, o similares para interpredicción. El codificador de video 20 y decodificador de video 30 también pueden ser compatibles con partición asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$, y $nR \times 2N$ para interpredicción.

[0156] La unidad de procesamiento de interpredicción 120 puede generar datos predictivos para una PU. Como parte de generar los datos predictivos para una PU, la unidad de procesamiento de interpredicción 120 realiza interpredicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos de la PU e información de movimiento para la PU. La unidad de procesamiento de interpredicción 120 puede realizar diferentes operaciones para una PU de una CU, lo que depende de si la PU está en un segmento I, un segmento P, o un segmento B. En un segmento I, todas las PU son intrapredicidas. Por ende, si la PU está en un segmento I, la unidad de procesamiento de interpredicción 120 no realiza interpredicción en la PU. Por lo tanto, para bloques codificados en modo I, el bloque de predicción se forma usando

predicción espacial de bloques vecinos previamente codificados dentro del mismo cuadro. Si una PU está en un segmento P, la unidad de procesamiento de interpredicción 120 puede usar interpredicción unidireccional para generar un bloque predictivo de la PU. Si una PU está en un segmento B, la unidad de procesamiento de interpredicción 120 puede usar interpredicción unidireccional o bidireccional para generar un bloque predictivo de la PU.

[0157] La unidad de procesamiento de intrapredicción 126 puede generar datos predictivos para una PU realizando intrapredicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos de la PU y diversos elementos sintácticos. La unidad de procesamiento de intrapredicción 126 puede realizar intrapredicción en PU en segmentos I, segmentos P y segmentos B.

[0158] Para realizar intrapredicción en una PU, la unidad de procesamiento de intrapredicción 126 puede usar múltiples modos de intrapredicción para generar múltiples conjuntos de datos predictivos para la PU. La unidad de procesamiento de intrapredicción 126 puede usar muestras de bloques de muestra de PU vecinas para generar un bloque predictivo para una PU. Las PU vecinas pueden estar arriba, arriba y a la derecha, arriba y a la izquierda, o a la izquierda de la PU, suponiendo un orden de codificación de izquierda a derecha, de arriba a abajo para las PU, CU y CTU. La unidad de procesamiento de intrapredicción 126 puede usar diversas cantidades de modos de intrapredicción, por ejemplo, 33 modos direccionales de intrapredicción. En algunos ejemplos, la cantidad de modos de intrapredicción puede depender del tamaño de la región asociada a la PU.

[0159] La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para las PU de una CU de entre los datos predictivos generados mediante la unidad de procesamiento de interpredicción 120 para las PU o los datos predictivos generados mediante la unidad de procesamiento de intrapredicción 126 para las PU. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos predictivos para las PU de la CU en función de medidas de tasa/distorsión de los conjuntos de datos predictivos. Los bloques predictivos de los datos predictivos seleccionados se pueden denominar en la presente bloques predictivos seleccionados.

[0160] La unidad de generación residual 102 puede generar, en función de los bloques de codificación (por ejemplo, bloques de codificación de luminancia, Cb y Cr) para una CU y los bloques predictivos seleccionados (por ejemplo, bloques predictivos de luminancia, Cb y Cr) para las PU de la CU, bloques residuales (por ejemplo, bloques residuales de luminancia, Cb y Cr) para la CU. Por ejemplo, la unidad de generación

residual 102 puede generar los bloques residuales de la CU tal que cada muestra en los bloques residuales tenga un valor igual a una diferencia entre una muestra en un bloque de codificación de la CU y una muestra correspondiente en un bloque predictivo seleccionado correspondiente de una PU de la CU.

[0161] La unidad de procesamiento de la transformada 104 puede particionar los bloques residuales de una CU en bloques de la transformada de las TU de la CU. Por ejemplo, la unidad de procesamiento de la transformada 104 puede realizar partición del árbol cuaternario para particionar los bloques residuales de la CU en bloques de la transformada de las TU de la CU. Por lo tanto, una TU se puede asociar a un bloque de la transformada de luminancia y dos bloques de la transformada de crominancia. Los tamaños y posiciones de los bloques de la transformada de luminancia y de crominancia de las TU de una CU pueden o no estar basados en los tamaños y posiciones de bloques de predicción de las PU de la CU. Una estructura de árbol cuaternario denominada un “árbol cuaternario residual” (RQT) puede incluir nodos asociados a cada una de las regiones. Las TU de una CU pueden corresponder a nodos hoja del RQT.

[0162] La unidad de procesamiento de la transformada 104 puede generar bloques de coeficiente de la transformada para cada TU de una CU aplicando una o más transformadas a los bloques de la transformada de la TU. La unidad de procesamiento de la transformada 104 puede aplicar diversas transformadas a un bloque de la transformada asociado a una TU. Por ejemplo, la unidad de procesamiento de la transformada 104 puede aplicar una transformada discreta del coseno (DCT), una transformada direccional, o una transformada conceptualmente similar a un bloque de la transformada. En algunos ejemplos, la unidad de procesamiento de la transformada 104 no aplica transformadas a un bloque de la transformada. En tales ejemplos, el bloque de la transformada se puede tratar como un bloque de coeficiente de la transformada.

[0163] La unidad de cuantificación 106 puede cuantificar los coeficientes de la transformada en un bloque de coeficientes. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o todos los coeficientes de la transformada. Por ejemplo, un coeficiente de la transformada de n bits se puede redondear para abajo hasta un coeficiente de la transformada de m bits durante la cuantificación, en donde n es mayor que m . La unidad de cuantificación 106 puede cuantificar un bloque de coeficientes asociado a una TU de una CU en función de un valor de parámetro de cuantificación (QP) asociado a la CU. El codificador de video 20 puede ajustar el grado

de cuantificación aplicado al bloque de coeficientes asociado a una CU mediante el ajuste del valor de QP asociado a la CU. La cuantificación puede provocar pérdida de información. Por lo tanto, los coeficientes de la transformada cuantificados pueden tener menor precisión que los originales.

[0164] La unidad de cuantificación inversa 108 y la unidad de procesamiento de la transformada inversa 110 pueden aplicar cuantificación inversa y transformadas inversas a un bloque de coeficientes, respectivamente, para reconstruir un bloque residual del bloque de coeficientes. La unidad de reconstrucción 112 puede agregar muestras del bloque residual reconstruido a las muestras correspondientes de uno o más bloques predictivos generados mediante una unidad de procesamiento de predicción 100 para producir un bloque reconstruido de la transformada asociado a una TU. Reconstruyendo bloques de la transformada para cada TU de una CU de esta manera, el codificador de video 20 puede reconstruir los bloques de codificación de la CU.

[0165] La unidad de filtro 114 puede realizar una o más operaciones de eliminación de bloques para reducir artefactos de compresión en los bloques de codificación asociados a una CU. La unidad de filtro 114 puede realizar las técnicas de filtro de la presente divulgación. Por ejemplo, la unidad de filtro 114 puede almacenar, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. En este ejemplo, cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal diferente respectiva. Además, en este ejemplo, la unidad de filtro 114 puede determinar, en función de un conjunto de parámetros de ALF seleccionado en una de las matrices que corresponden a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En este ejemplo, la unidad de filtro 114 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a uno o más bloques en la región actual.

[0166] En otro ejemplo, la unidad de filtro 114 puede almacenar, en una matriz (por ejemplo, matriz 70 de la Figura 6), conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual. De forma adicional, la unidad de filtro 114 puede almacenar, en la matriz, índices de capa temporal asociados con los conjuntos de parámetros de ALF. Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de

parámetros de ALF se usó para aplicar un filtro de ALF. Además, en este ejemplo, la unidad de filtro 114 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual o una capa temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En este ejemplo, la unidad de filtro 114 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a uno o más bloques en la región actual.

[0167] La memoria de imagen decodificada 116 puede almacenar los bloques de codificación reconstruidos luego de que la unidad de filtro 114 realiza la una o más operaciones de eliminación de bloques en los bloques de codificación reconstruidos. La unidad de procesamiento de interpredicción 120 puede usar una imagen de referencia que contiene los bloques de codificación reconstruidos para realizar interpredicción en las PU de otras imágenes. Además, la unidad de procesamiento de intrapredicción 126 puede usar bloques de codificación reconstruidos en la memoria de imagen decodificada 116 para realizar intrapredicción en otras PU en la misma imagen que la CU.

[0168] La unidad de codificación entrópica 118 puede recibir datos de otros componentes funcionales del codificador de video 20. Por ejemplo, la unidad de codificación entrópica 118 puede recibir bloques de coeficientes de la unidad de cuantificación 106 y puede recibir elementos sintácticos de la unidad de procesamiento de predicción 100. La unidad de codificación entrópica 118 puede realizar una o más operaciones de codificación entrópica sobre los datos para generar datos de codificación entrópica. Por ejemplo, la unidad de codificación entrópica 118 puede realizar una operación CABAC, una operación de codificación adaptativa según el contexto de longitud variable (CAVLC), una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética binaria adaptativa según el contexto en función de la sintaxis (SBAC), una operación de codificación entrópica de partición en intervalos de probabilidad (PIPE), una operación de codificación Exponencial-Golomb, u otro tipo de operación de codificación entrópica sobre datos. El codificador de video 20 puede enviar una corriente de bits que incluye datos codificados por entropía generados mediante la unidad de codificación entrópica 118. Por ejemplo, la corriente de bits puede incluir datos que representan valores de coeficientes de la transformada para una CU.

[0169] La Figura 8 es un diagrama de bloques que ilustra un decodificador de video 30

de ejemplo que se configura para implementar las técnicas de la presente divulgación. La Figura 8 se proporciona con fines explicativos y no es limitante de las técnicas ampliamente ejemplificadas y descritas en la presente divulgación. Con fines explicativos, la presente divulgación describe el decodificador de video 30 en el contexto de la codificación HEVC. Sin embargo, las técnicas de la presente divulgación pueden ser aplicables a otros métodos o estándares de codificación.

[0170] El circuito de procesamiento incluye el decodificador de video 30, en donde el decodificador de video 30 se configura para realizar una o más de las técnicas de ejemplo descritas en la presente divulgación. Por ejemplo, el decodificador de video 30 incluye un circuito integrado, y las diversas unidades ilustradas en la Figura 8 se pueden formar como bloques de circuitos de *hardware* que están interconectados con un bus de circuito. Estos bloques de circuitos de *hardware* pueden ser bloques de circuitos separados o dos o más de las unidades se pueden combinar en un bloque de circuitos de *hardware* común. Los bloques de circuitos de *hardware* se pueden formar como combinación de componentes eléctricos que forman bloques de operación tales como unidades lógicas aritméticas (ALU), unidades de función elemental (EFU), así como bloques lógicos tales como AND, OR, NAND, NOR, XOR, XNOR y otros bloques lógicos similares.

[0171] En algunos ejemplos, una o más de las unidades ilustradas en la Figura 8 pueden ser unidades de *software* que se ejecutan en el circuito de procesamiento. En tales ejemplos, el código de objeto para estas unidades de *software* se almacena en la memoria. Un sistema operativo puede ocasionar que el decodificador de video 30 recupere el código de objeto y ejecute el código de objeto, que hace que el decodificador de video 30 realice operaciones para implementar las técnicas de ejemplo. En algunos ejemplos, las unidades de *software* pueden ser *firmware* que el decodificador de video 30 ejecuta al encenderse. En consecuencia, el decodificador de video 30 es un componente estructural que tiene *hardware* que realiza las técnicas de ejemplo o tiene *software/firmware* que se ejecuta en el *hardware* para especializar el hardware para realizar las técnicas de ejemplo.

[0172] En el ejemplo de la Figura 8, el decodificador de video 30 incluye una unidad de decodificación entrópica 150, memoria de datos de video 151, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de procesamiento de la transformada inversa 156, una unidad de reconstrucción 158, una unidad de filtro 160, y una memoria de imagen decodificada 162. La unidad de procesamiento de predicción 152 incluye una unidad de

compensación de movimiento 164 y una unidad de procesamiento de intrapredicción 166. En otros ejemplos, el decodificador de video 30 puede incluir más, menos, o diferentes componentes funcionales.

[0173] La memoria de datos de video 151 puede almacenar datos de video codificados, tales como una corriente de bits de video codificada, para decodificar mediante los componentes del decodificador de video 30. Los datos de video almacenados en la memoria de datos de video 151 se pueden obtener, por ejemplo, de un medio legible por computadora 16, por ejemplo, de una fuente de video local, tal como una cámara, a través de una comunicación en red inalámbrica o alámbrica de datos de video, o accediendo a medios de almacenamiento de datos físicos. La memoria de datos de video 151 puede formar una memoria de imagen codificada (CPB) que almacena datos de video codificados de una corriente de bits de video codificada. La memoria de imagen decodificada 162 puede ser una memoria de imagen de referencia que almacena datos de video de referencia para usar en la decodificación de datos de video mediante el decodificador de video 30, por ejemplo, en modos de intra- o intercodificación, o para enviar. La memoria de datos de video 151 y la memoria de imagen decodificada 162 se pueden formar con una variedad de dispositivos de memoria, tal como memoria dinámica de acceso aleatorio (DRAM), que incluye DRAM síncrona (SDRAM), RAM magnetoresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de video 151 y la memoria de imagen decodificada 162 se pueden proporcionar mediante el mismo dispositivo de memoria o mediante dispositivos de memoria diferentes. En diversos ejemplos, la memoria de datos de video 151 puede ser en chip con otros componentes del decodificador de video 30 o externa con respecto a aquellos componentes. La memoria de datos de video 151 puede ser la misma que o parte de los medios de almacenamiento 28 de la Figura 1.

[0174] La memoria de datos de video 151 recibe y almacena datos de video codificados (por ejemplo, unidades de NAL) de una corriente de bits. La unidad de decodificación entrópica 150 puede recibir datos de video codificados (por ejemplo, unidades de NAL) de memoria de datos de video 151 y puede analizar las unidades de NAL para obtener elementos sintácticos. La unidad de decodificación entrópica 150 puede decodificar por entropía los elementos sintácticos codificados por entropía en las unidades de NAL. La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de procesamiento de la transformada inversa 156, la unidad de reconstrucción 158, y unidad de filtro 160 pueden generar datos de

video decodificados en función de los elementos sintácticos extraídos de la corriente de bits. La unidad de decodificación entrópica 150 puede realizar un proceso generalmente recíproco al de la unidad de codificación entrópica 118.

[0175] Además de obtener elementos sintácticos de la corriente de bits, el decodificador de video 30 puede realizar una operación de reconstrucción en una CU sin particionar. Para realizar la operación de reconstrucción en una CU, el decodificador de video 30 puede realizar una operación de reconstrucción en cada TU de la CU. Realizando la operación de reconstrucción para cada TU de la CU, el decodificador de video 30 puede reconstruir bloques residuales de la CU.

[0176] Como parte de realizar una operación de reconstrucción en una TU de una CU, la unidad de cuantificación inversa 154 puede cuantificar de manera inversa, es decir, descuantificar, bloques de coeficientes asociados a la TU. Luego de que la unidad de cuantificación inversa 154 cuantifica de manera inversa un bloque de coeficientes, la unidad de procesamiento de la transformada inversa 156 puede aplicar una o más transformadas inversas al bloque de coeficientes para generar un bloque residual asociado a la TU. Por ejemplo, la unidad de procesamiento de la transformada inversa 156 puede aplicar una DCT inversa, una transformada entera inversa, una transformada de Karhunen-Loeve (KLT) inversa, una transformada rotacional inversa, una transformada direccional inversa, u otra transformada inversa al bloque de coeficientes.

[0177] La unidad de cuantificación inversa 154 puede realizar técnicas particulares de la presente divulgación. Por ejemplo, para al menos un grupo de cuantificación respectivo de una pluralidad de grupos de cuantificación dentro de un CTB de una CTU de una imagen de los datos de video, la unidad de cuantificación inversa 154 puede derivar, en base al menos en parte en información local de cuantificación señalizada en la corriente de bits, un parámetro respectivo de cuantificación para el grupo de cuantificación respectivo. Además, en este ejemplo, la unidad de cuantificación inversa 154 puede cuantificar de manera inversa, en función del parámetro de cuantificación respectivo para el grupo de cuantificación respectivo, al menos un coeficiente de la transformada de un bloque de la transformada de una TU de una CU de la CTU. En este ejemplo, el grupo de cuantificación respectivo se define como un grupo de CU o bloques de codificación sucesivos, en orden de codificación, por lo que los límites del grupo de cuantificación respectivo deben ser límites de las CU o bloques de codificación y un tamaño del grupo de cuantificación respectivo es mayor que o igual a un umbral. El decodificador de video 30 (por ejemplo, la unidad de

procesamiento de la transformada inversa 156, la unidad de reconstrucción 158, y unidad de filtro 160) puede reconstruir, en función de coeficientes de la transformada cuantificados de manera inversa del bloque de la transformada, un bloque de codificación de la CU.

[0178] Si una PU se codifica usando intrapredicción, la unidad de procesamiento de intrapredicción 166 puede realizar intrapredicción para generar bloques predictivos de la PU. La unidad de procesamiento de intrapredicción 166 puede usar un modo de intrapredicción para generar los bloques predictivos de la PU en función de muestras de bloques vecinos espaciales. La unidad de procesamiento de intrapredicción 166 puede determinar el modo de intrapredicción para la PU en función de uno o más elementos sintácticos obtenidos de la corriente de bits.

[0179] Si una PU se codifica usando interpredicción, la unidad de decodificación entrópica 150 puede determinar información de movimiento para la PU. La unidad de compensación de movimiento 164 puede determinar, en función de la información de movimiento de la PU, uno o más bloques de referencia. La unidad de compensación de movimiento 164 puede generar, en función de el uno o más bloques de referencia, bloques predictivos (por ejemplo, bloques predictivos de luminancia, Cb y Cr) para la PU.

[0180] La unidad de reconstrucción 158 puede usar bloques de la transformada (por ejemplo, bloques de la transformada de luminancia, Cb y Cr) para las TU de una CU y los bloques predictivos (por ejemplo, bloques de luminancia, Cb y Cr) de las PU de la CU, es decir, datos de intrapredicción o datos de interpredicción, según corresponda, para reconstruir los bloques de codificación (por ejemplo, bloques de codificación de luminancia, Cb y Cr) para la CU. Por ejemplo, la unidad de reconstrucción 158 puede agregar muestras de los bloques de la transformada (por ejemplo, bloques de la transformada de luminancia, Cb y Cr) a muestras correspondientes de los bloques predictivos (por ejemplo, bloques predictivos de luminancia, Cb y Cr) para reconstruir los bloques de codificación (por ejemplo, bloques de codificación de luminancia, Cb y Cr) de la CU.

[0181] La unidad de filtro 160 puede realizar una operación de eliminación de bloques para reducir artefactos de compresión asociados a los bloques de codificación de la CU. La unidad de filtro 160 puede realizar las técnicas de filtro de la presente divulgación. Por ejemplo, la unidad de filtro 160 puede almacenar, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que una imagen

actual. En este ejemplo, cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal distinta respectiva. Por ejemplo, para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, la unidad de filtro 160 puede almacenar, en la matriz respectiva, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video que son decodificados antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva. En este ejemplo, la unidad de filtro 160 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz que corresponde a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. Además, en este ejemplo, la unidad de filtro 160 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, un filtro de ALF a uno o más bloques en la región actual.

[0182] En otro ejemplo, la unidad de filtro 160 almacena, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que una imagen actual. Además, en este ejemplo, la unidad de filtro 160 almacena, en la matriz, índices de capa temporal asociados con los conjuntos de parámetros de ALF. Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF. En este ejemplo, la unidad de filtro 160 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual. En este ejemplo, la unidad de filtro 160 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, un filtro de ALF a uno o más bloques de la región actual.

[0183] El decodificador de video 30 puede almacenar los bloques de codificación de la CU en la memoria de imagen decodificada 162. La memoria de imagen decodificada 162 puede proporcionar imágenes de referencia para la posterior compensación de movimiento, intrapredicción, y presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la Figura 1. Por ejemplo, el decodificador de video 30 puede realizar, en función de los bloques en la memoria de imagen decodificada 162, operaciones de intrapredicción o interpredicción para las PU de otras CU.

[0184] Se han descrito ciertos aspectos de la presente divulgación con respecto a

extensiones de los estándares de HEVC con fines ilustrativos. Sin embargo, las técnicas descritas en la presente divulgación pueden ser útiles para otros procesos de codificación de video, que incluyen otros procesos de codificación de video estándares o propietarios aún no desarrollados.

[0185] La Figura 9 es un diagrama de flujo que ilustra una operación de ejemplo del codificador de video 20, de acuerdo con la primera técnica de la presente divulgación. Los diagramas de flujo de la presente divulgación se proporcionan como ejemplos. En otros ejemplos, se pueden realizar acciones en diferentes órdenes, o las operaciones pueden incluir más, menos, o diferentes acciones.

[0186] En el ejemplo de la Figura 9, el codificador de video 20 genera una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video (200). Una región actual (por ejemplo, un segmento actual u otra unidad) de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual. El codificador de video 20 puede generar la corriente de bits de acuerdo con cualquiera de los ejemplos descritos en otras partes de la presente divulgación, tales como el ejemplo de la Figura 5.

[0187] Además, el codificador de video 20 reconstruye la imagen actual (202). Por ejemplo, el codificador de video 20 puede reconstruir un bloque de la imagen actual agregando muestras de bloques residuales reconstruidos a muestras correspondientes de uno o más bloques predictivos para producir bloques reconstruidos. Mediante la reconstrucción de bloques de esta manera, el codificador de video 20 puede reconstruir los bloques de codificación de la imagen actual.

[0188] Además, para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, el codificador de video 20 puede almacenar, en la matriz respectiva (por ejemplo, una de las matrices 60 de la Figura 5), conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video que son decodificados antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva (204). Cada conjunto de parámetros de ALF puede incluir un conjunto de coeficientes de filtro y/o un conjunto de información de fusión de clases de ALF.

[0189] El codificador de video 20 determina, en función de un conjunto de parámetros de ALF seleccionado en una de las matrices que corresponden a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual (206). Por ejemplo, el codificador de video 20 puede seleccionar el

conjunto de parámetros de ALF seleccionado usando un análisis de tasa-distorsión de los conjuntos de parámetros de ALF en la matriz que corresponden a la capa temporal a la que pertenece la región actual. En algunos ejemplos, el conjunto de parámetros de ALF aplicable para la región actual puede ser el mismo que el conjunto de parámetros de ALF seleccionado. En algunos ejemplos, el codificador de video 20 puede incluir, en la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual.

[0190] Luego, el codificador de video 20 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual (208). Cuando se aplica filtrado en bucle adaptativo a la región actual, el codificador de video 20 puede aplicar un filtro de ALF a uno o más, pero no necesariamente todos, de los bloques en la región actual. Por ejemplo, el codificador de video 20 puede dividir la región actual en bloques (por ejemplo, bloques 4x4). En este ejemplo, para cada uno de los bloques, el codificador de video 20 puede determinar (por ejemplo, en función de una dirección y actividad del bloque) una categoría correspondiente para el bloque. En este ejemplo, el conjunto de parámetros de ALF aplicable para la región actual puede incluir coeficientes de filtro de un filtro de ALF de la categoría para el bloque. En este ejemplo, el codificador de video 20 puede luego aplicar el filtro de ALF al bloque.

[0191] Luego de aplicar el filtrado en bucle adaptativo a la región actual, el codificador de video 20 usa la región actual para predicción de una imagen posterior de los datos de video (210). Por ejemplo, el codificador de video 20 puede usar la región actual para interpredicción de un bloque de la imagen posterior.

[0192] La Figura 10 es un diagrama de flujo que ilustra una operación de ejemplo del decodificador de video 30, de acuerdo con la primera técnica de la presente divulgación. En el ejemplo de la Figura 10, el decodificador de video 30 recibe una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video (250). Una región actual de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual.

[0193] Además, el decodificador de video 30 reconstruye la imagen actual (252). El decodificador de video 30 puede reconstruir la imagen actual de acuerdo con cualquiera de los ejemplos proporcionados en otras partes de la presente divulgación. Por ejemplo, el decodificador de video 30 puede reconstruir un bloque de la imagen actual agregando muestras de bloques residuales reconstruidos a muestras

correspondientes de uno o más bloques predictivos para producir bloques reconstruidos. Reconstruyendo bloques de esta manera, el decodificador de video 30 puede reconstruir los bloques de codificación de la imagen actual.

[0194] El decodificador de video 30 también almacena, en una pluralidad de matrices, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual (254). Cada matriz respectiva de la pluralidad de matrices corresponde a una capa temporal distinta respectiva. Por ejemplo, para cada matriz respectiva de una pluralidad de matrices que corresponden a diferentes capas temporales, el decodificador de video 30 puede almacenar, en la matriz respectiva (por ejemplo, una de las matrices 60 de la Figura 5), conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de regiones de imágenes de los datos de video que son decodificados antes que la región actual y que están en la capa temporal que corresponde a la matriz respectiva o una capa temporal inferior a la capa temporal que corresponde a la matriz respectiva.

[0195] Además, el decodificador de video 30 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz que corresponde a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual (256). Por ejemplo, el decodificador de video 30 puede obtener, de la corriente de bits, un índice que indica el conjunto seleccionado de parámetros de ALF en la matriz que corresponde a la capa temporal a la que pertenece la región actual. En algunos ejemplos, el conjunto de parámetros de ALF aplicable para la región actual puede ser el mismo que el conjunto de parámetros de ALF seleccionado. En algunos ejemplos, el decodificador de video 30 puede obtener, de la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual.

[0196] Luego, el decodificador de video 30 puede aplicar, en función del conjunto de parámetros de ALF aplicable para la región actual, filtrado en bucle adaptativo a la región actual (258). Cuando se aplica filtrado en bucle adaptativo a la región actual, el decodificador de video 30 puede aplicar un filtro de ALF a uno o más, pero no necesariamente todos, de los bloques en la región actual. Por ejemplo, el decodificador de video 30 puede dividir la región actual en bloques (por ejemplo, bloques 4x4). En este ejemplo, para cada uno de los bloques, el decodificador de video 30 puede determinar (por ejemplo, en función de una dirección y actividad del

bloque) una categoría correspondiente para el bloque. En este ejemplo, el conjunto de parámetros de ALF aplicable para la región actual puede incluir coeficientes de filtro de un filtro de ALF de la categoría para el bloque. En este ejemplo, el decodificador de video 30 puede luego aplicar el filtro de ALF al bloque.

[0197] La Figura 11 es un diagrama de flujo que ilustra una operación de ejemplo del codificador de video 20 de acuerdo con la segunda técnica de la presente divulgación. En el ejemplo de la Figura 11, el codificador de video 20 genera una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video (300). Una región actual (por ejemplo, un segmento actual u otra unidad) de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual. El codificador de video 20 puede generar la corriente de bits de acuerdo con cualquiera de los ejemplos descritos en otras partes de la presente divulgación, tales como el ejemplo de la Figura 5.

[0198] Además, el codificador de video 20 puede reconstruir la imagen actual (302). Por ejemplo, el codificador de video 20 puede reconstruir un bloque de la imagen actual agregando muestras de bloques residuales reconstruidos a muestras correspondientes de uno o más bloques predictivos para producir bloques reconstruidos. Mediante la reconstrucción de bloques de esta manera, el codificador de video 20 puede reconstruir los bloques de codificación de la imagen actual.

[0199] El codificador de video 20 almacena, en una matriz (por ejemplo, matriz 70 de la Figura 6), conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual (304). Además, el codificador de video 20 almacena, en la matriz, índices de capa temporal asociados a los conjuntos de parámetros de ALF (306). Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF.

[0200] Además, el codificador de video 20 determina, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual o una capa temporal inferior a la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual (308). Por ejemplo, el codificador de video 20 puede seleccionar el conjunto de parámetros de ALF seleccionado usando un análisis de tasa-distorsión de los conjuntos de parámetros de ALF en la matriz. En algunos ejemplos, el conjunto de parámetros de ALF aplicable para la región actual

puede ser el mismo que el conjunto de parámetros de ALF seleccionado. En algunos ejemplos, el codificador de video 20 puede incluir, en la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual.

[0201] El codificador de video 20 aplica, en función del conjunto de parámetros de ALF aplicable para la región actual, un ALF a la región actual (310). El codificador de video 20 puede aplicar el filtro de ALF a la región actual de acuerdo con cualquiera de los ejemplos proporcionados en otras partes de la presente divulgación.

[0202] Luego de aplicar el filtro de ALF a la región actual, el codificador de video 20 usa la región actual para predicción de una imagen posterior de los datos de video (312). Por ejemplo, el codificador de video 20 puede usar la región actual para interpredicción de un bloque de la imagen posterior.

[0203] La Figura 12 es un diagrama de flujo que ilustra una operación de ejemplo de un decodificador de video 30 de acuerdo con una técnica de la presente divulgación. En el ejemplo de la Figura 12, el decodificador de video 30 recibe una corriente de bits que incluye una representación codificada de una imagen actual de los datos de video (350). Una región actual (por ejemplo, un segmento actual u otra unidad) de la imagen actual se asocia a un índice temporal que indica una capa temporal a la que pertenece la región actual.

[0204] Luego, el decodificador de video 30 puede reconstruir la imagen actual (352). El decodificador de video 30 puede reconstruir la imagen actual de acuerdo con cualquiera de los ejemplos proporcionados en otras partes de la presente divulgación. Por ejemplo, el decodificador de video 30 puede reconstruir un bloque de la imagen actual agregando muestras de bloques residuales reconstruidos a muestras correspondientes de uno o más bloques predictivos para producir bloques reconstruidos. Reconstruyendo bloques de esta manera, el decodificador de video 30 puede reconstruir los bloques de codificación de la imagen actual.

[0205] En el ejemplo de la Figura 12, el decodificador de video 30 almacena, en una matriz, conjuntos de parámetros de ALF usados en la aplicación de filtros de ALF a muestras de imágenes de los datos de video decodificados antes que la imagen actual (354). Además, el decodificador de video 30 almacena, en la matriz, índices de capa temporal asociados a los conjuntos de parámetros de ALF (356). Un índice de capa temporal asociado a un conjunto de parámetros de ALF indica una capa temporal de una región en la que el conjunto de parámetros de ALF se usó para aplicar un filtro de ALF.

[0206] El decodificador de video 30 puede determinar, en función de un conjunto seleccionado de parámetros de ALF en la matriz cuyo índice de capa temporal asociado indica la capa temporal a la que pertenece la región actual, un conjunto de parámetros de ALF aplicable para la región actual (358). Por ejemplo, el decodificador de video 30 puede obtener, de la corriente de bits, un índice que indica el conjunto seleccionado de parámetros de ALF en la matriz. En algunos ejemplos, el conjunto de parámetros de ALF aplicable para la región actual puede ser el mismo que el conjunto de parámetros de ALF seleccionado. En algunos ejemplos, el decodificador de video 30 puede obtener, de la corriente de bits, una indicación de una diferencia entre el conjunto de parámetros de ALF seleccionado y el conjunto de parámetros de ALF aplicable para la región actual.

[0207] Luego, el decodificador de video 30 aplica, en función del conjunto de parámetros de ALF aplicable para la región actual, un filtro de ALF a la región actual (360). El decodificador de video 30 puede aplicar el filtro de ALF a la región actual de acuerdo con cualquiera de los ejemplos proporcionados en otras partes de la presente divulgación.

[0208] Un codificador de video, como se describió en la presente divulgación, se puede referir a un codificador de video o a un decodificador de video. De manera similar, una unidad de codificación de video se puede referir a un codificador de video o a un decodificador de video. Asimismo, la codificación de video se puede referir tanto a la codificación como a la decodificación de video, según corresponda. En la presente divulgación, la expresión “en función de” puede indicar basado solo en, basado al menos en parte en, o basado de alguna manera en. La presente divulgación puede usar la expresión “unidad de video” o “bloque de video” o “bloque” para referirse a uno o más bloques de muestra y estructuras sintácticas usadas para codificar muestras del uno o más bloques de muestras. Los tipos de unidades de video de ejemplo pueden incluir CTU, CU, PU, unidades de la transformada (TU), macrobloques, particiones de macrobloques, etcétera. En algunos contextos, la discusión de las PU puede ser indistinta de la discusión de macrobloques o particiones de macrobloques. Los tipos de bloques de video de ejemplo pueden incluir bloques de codificación en árbol, bloques de codificación y otros tipos de bloques de datos de video.

[0209] 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.

[0210] 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.

[0211] 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 computadora pueden incluir medios de almacenamiento legibles por computadora, lo que corresponde a un medio tangible, tal como medios de almacenamiento de datos o medios de comunicación que incluyen cualquier medio que facilita la transferencia de un programa de computadora desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por computadora generalmente pueden corresponder a (1) un medio de almacenamiento tangible legible por computadora que no sea transitorio o (2) un medio de comunicación, tal como una señal o una onda 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 circuitos de procesamiento para obtener instrucciones, códigos y/o estructuras de datos para la implementación de las técnicas descritas en la presente divulgación. Un producto de programa de computación puede incluir un medio legible por computadora.

[0212] 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 conoce con el término medio legible por computadora. Por ejemplo, si las

instrucciones son transmitidas desde una página web, servidor u otra fuente remota que utilice un cable coaxial, cable de fibra óptica, par trenzado, línea de abonado digital (DSL), o tecnologías inalámbricas tales como infrarrojo, radio y microondas, entonces el cable coaxial, cable de fibra óptica, par trenzado, DSL, o tecnologías inalámbricas tales como infrarrojo, radio, y microondas se incluyen 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. Disco y diskette, como se usa en la presente, incluye disco compacto (CD), disco láser, disco óptico, disco digital versátil (DVD), diskette y disco Blu-ray, en donde los diskettes reproducen datos en forma magnética, mientras que los discos reproducen datos en forma óptica con láser. También deberían incluirse combinaciones de lo anterior dentro del alcance de medios legibles por computadora.

[0213] La funcionalidad descrita en la presente divulgación se puede realizar mediante circuitos de procesamiento de función fija y/o programables. Por ejemplo, las instrucciones se pueden ejecutar mediante circuitos de procesamiento de función fija y/o programables. Tales circuitos de procesamiento pueden incluir 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 lógicas programables en campo (FPGA) u otro circuito lógico integrado o separado equivalente. En consecuencia, como se usa en la presente, el término “procesador” puede referirse a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en la presente. Además, en algunos aspectos, la funcionalidad descrita en la presente puede estar proporcionada dentro de módulos de *hardware* y/o *software* dedicados configurados para codificar y decodificar, o puede estar incorporada en un códec combinado. Además, las técnicas podrían implementarse totalmente en uno o más circuitos o elementos lógicos. Los circuitos de procesamiento se pueden acoplar a otros componentes de diversas formas. Por ejemplo, un circuito de procesamiento se puede acoplar a otros componentes mediante una interconexión del dispositivo interna, una conexión de red alámbrica o inalámbrica u otro medio de comunicación.

[0214] Las técnicas de la presente divulgación 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 divulgación se describen distintos componentes, módulos o unidades para

enfaticar los aspectos funcionales de los dispositivos configurados para realizar las técnicas descritas, pero no es necesario que sean realizadas por diferentes unidades de *hardware*. En su lugar, como se describió anteriormente, distintas unidades pueden combinarse en una unidad de *hardware* de códec o pueden ser proporcionadas por un grupo de unidades de *hardware* interoperativas, que incluyen uno o más procesadores, como se describió anteriormente, junto con *software* y/o *firmware* adecuado.

[0215] Se han descrito distintos ejemplos. Estos y otros ejemplos se encuentran dentro del alcance de las siguientes reivindicaciones.