

REMUESTREO DE IMÁGENES DE REFERENCIA CON FILTROS CONMUTABLES

CAMPO

- 5 [0001] La presente solicitud se refiere a la codificación y compresión de video. Más específicamente, la presente solicitud se refiere a sistemas y métodos para proporcionar remuestreo de imágenes de referencia con filtros conmutables.

ANTECEDENTES

- 10 [0002] Muchos dispositivos y sistemas permiten procesar y emitir datos de video para consumo. Los datos de video digital incluyen grandes cantidades de datos para satisfacer las demandas de los consumidores y proveedores de video. Por ejemplo, los consumidores de datos de video desean video de la máxima calidad, con alta fidelidad, resoluciones, tasas de cuadros y similares. Como resultado, la gran cantidad de datos de video que se requieren para satisfacer estas demandas supone una carga para las
- 15 redes y dispositivos de comunicación que procesan y almacenan los datos de video.

- [0003] Pueden usarse varias técnicas de codificación de video para comprimir datos de video. La codificación de video se realiza de acuerdo con uno o más estándares de codificación de video. Por ejemplo, los estándares de codificación de video incluyen codificación de video de alta eficiencia (HEVC), codificación de video avanzada (AVC),
- 20 codificación del grupo de expertos en imágenes en movimiento (MPEG) 2 parte 2, VP9, Alliance of Open Media (AOMedia) Video 1 (AV1), Codificación de video esencial (EVC), o similar. La codificación de video generalmente utiliza métodos de predicción (por ejemplo, interpredicción, intrapredicción o similares) que aprovechan la redundancia presente en las imágenes o secuencias de video. Un objetivo importante de las técnicas
- 25 de codificación de video es comprimir los datos de video en una forma que utiliza una tasa de bits más baja, mientras que se evita o minimiza la degradación de la calidad del video. Con la disponibilidad de servicios de video en constante evolución, se necesitan técnicas de codificación con una mejor eficiencia de codificación.

- 30 **BREVE SÍNTESIS**

- [0004] Se describen aquí sistemas y métodos para proporcionar técnicas mejoradas de procesamiento de video. Algunos ejemplos de procesamiento de video utilizan técnicas de compresión de video con predicción para codificar y decodificar datos de video de manera eficiente. Por ejemplo, un codificador de video puede realizar predicciones para
- 35 un bloque actual que se está codificando comparando valores de píxeles en el bloque

actual con un bloque de predicción (por ejemplo, un bloque que no ha sido codificado o que ha sido decodificado previamente). Un error de predicción (también denominado residual) se puede determinar como una diferencia entre el bloque actual y el bloque de predicción. Cuando se realiza la predicción para un bloque en una imagen usando uno o más bloques de una o más imágenes (lo que se conoce como interpredicción), se puede usar un vector de movimiento para indicar una ubicación de un bloque de predicción en relación con el bloque actual. El codificador de video puede aplicar una transformada al error de predicción (por ejemplo, una transformada de coseno discreta (DCT) u otra transformada adecuada) para generar coeficientes de transformada.

Después de la transformación, el codificador de video puede cuantificar los coeficientes de transformada. Los coeficientes de transformada cuantificados y los vectores de movimiento pueden representarse utilizando elementos de sintaxis y, junto con la información de control, formar una representación codificada de una secuencia de video. En algunos casos, el codificador de video puede entropizar elementos de sintaxis de código, lo que puede reducir aún más el número de bits necesarios para representar los elementos de sintaxis.

[0005] En algunos ejemplos, la conversión de frecuencia de muestreo se puede aplicar a datos de video. La conversión de frecuencia de muestreo puede incluir remuestreo, submuestreo, sobremuestreo, interpolación y/u otras técnicas de conversión de frecuencia de muestreo. Por ejemplo, la interpolación puede incluir interpolación constante por partes, interpolación lineal, interpolación polinomial e interpolación spline, entre otras técnicas de interpolación. El remuestreo puede incluir submuestreo, sobremuestreo y/o una combinación de submuestreo y sobremuestreo, o la obtención de nuevas muestras a una frecuencia de muestreo diferente.

[0006] En algunos ejemplos, la predicción de movimiento se puede mejorar con una técnica de fusión de bloques de interpredicción para simplificar la señalización de datos de movimiento por bloques. Las tareas de interpolación se utilizan en dicho contexto debido a que el desplazamiento de objetos entre imágenes (por ejemplo, entre una imagen actual y una imagen de referencia) es independiente de la cuadrícula de píxeles de las imágenes. La precisión fraccionaria (por ejemplo, la estimación de la ubicación entre píxeles en las imágenes) se puede utilizar para capturar el movimiento con mayor precisión. Los filtros de interpolación se pueden usar en el procesamiento de video para implementar una precisión fraccionada, particularmente cuando la imagen de referencia y la imagen actual que se procesa comparten el mismo tamaño. Los filtros de interpolación se utilizan para filtrar datos de imagen para estimar valores de imagen en

posiciones fraccionarias, y estos datos de posición fraccionaria se pueden utilizar con predicción u otras operaciones de procesamiento de imágenes.

[0007] El remuestreo se refiere a tomar una imagen con ciertas muestras (por ejemplo, píxeles en una determinada cuadrícula) y modificar las muestras (por ejemplo, creando nuevos píxeles en una cuadrícula diferente) mientras se mantienen los datos de las muestras originales. Como se señaló con anterioridad, el remuestreo puede incluir el sobremuestreo, en donde los píxeles de una cuadrícula de menor resolución se utilizan para crear píxeles en una cuadrícula de mayor resolución, así como el submuestreo, en donde los píxeles de una cuadrícula de mayor resolución se utilizan para generar píxeles en una cuadrícula de menor resolución. En el contexto de la predicción de movimiento y los vectores de movimiento descritos con anterioridad, cuando la imagen de referencia y la imagen actual que se está comparando son de diferentes tamaños, el remuestreo de la imagen de referencia se puede utilizar para estandarizar la cuadrícula de píxeles como parte de la generación de una imagen con precisión fraccionaria. Cuando un sistema de procesamiento de imágenes está configurado para permitir imágenes de referencia con un tamaño diferente al de una imagen actual, un filtro indicado por un índice de filtro puede ser adecuado para procesar las imágenes para el filtrado de interpolación (por ejemplo, precisión fraccionaria cuando la referencia es del mismo tamaño que una imagen actual) pero no para precisión fraccionaria con remuestreo de imagen de referencia (por ejemplo, precisión fraccionaria cuando la imagen de referencia es de un tamaño diferente al de la imagen actual). En algunos sistemas, por ejemplo, un filtro de suavizado usado para precisión fraccionaria puede generar buenos resultados con interpolación cuando la imagen de referencia y la imagen actual son del mismo tamaño, pero puede generar resultados deficientes cuando se usa con remuestreo de imagen de referencia donde la imagen de referencia tiene una cuadrícula de píxeles diferente (por ejemplo, un tamaño diferente) que la imagen actual. Los ejemplos descritos en el presente documento mejoran el funcionamiento de los dispositivos de decodificación y codificación identificando valores de filtro apropiados y generando resultados de codificación y decodificación mejorados durante las operaciones de los dispositivos de codificación y decodificación.

[0008] En varios ejemplos, tales mejoras se pueden implementar identificando información de índice de filtro para un bloque actual de la imagen actual y determinando que un primer valor de tamaño de imagen de la imagen actual y un segundo valor de tamaño de imagen de la imagen de referencia son diferentes (por ejemplo, cuando se utiliza remuestreo de imagen de referencia). En este contexto, el ejemplo realiza un

proceso de remuestreo usando un índice de filtro predeterminado en lugar de un índice de filtro actual identificado por la información del índice de filtro (por ejemplo, sin usar el índice de filtro actual).

5 **[0009]** La presente revelación provee un método de procesamiento de datos de video de acuerdo con la reivindicación 1 y un aparato para procesar datos de video de acuerdo con la reivindicación 14. Realizaciones específicas se definen en las reivindicaciones dependientes.

10 **[0010]** En algunos aspectos, el aparato para procesar datos de video puede incluir un dispositivo o aparato de decodificación y/o un dispositivo o aparato de codificación. El aparato para procesar datos de video puede ser una cámara, un dispositivo móvil (por ejemplo, un teléfono móvil o el llamado “teléfono inteligente” u otro dispositivo móvil), un dispositivo portátil, un dispositivo de realidad extendida (por ejemplo, un dispositivo de realidad virtual (VR), un dispositivo de realidad aumentada (AR) o un dispositivo de realidad mixta (MR)), una computadora personal, una computadora portátil, una
15 computadora servidor u otro dispositivo. En algunos aspectos, el aparato para procesar datos de video incluye una cámara o múltiples cámaras para capturar una o más imágenes. En algunos aspectos, el aparato para procesar datos de video incluye una pantalla para mostrar una o más imágenes, notificaciones y/u otros datos visualizables.

20 **[0011]** Los aspectos descritos con anterioridad relacionados con cualquiera de los métodos, aparatos y medios legibles por computadora pueden usarse individualmente o en cualquier combinación adecuada.

25 **[0012]** Este resumen no está destinado a identificar características clave o esenciales del objeto reivindicado, ni está destinado a ser utilizado de forma aislada para determinar el alcance del objeto reivindicado. El objeto debe entenderse por referencia a las porciones apropiadas de la memoria descriptiva completa de esta patente, cualquiera o todos los dibujos, y cada reivindicación.

30 **[0013]** Lo anterior, junto con otras características y formas de realización, se hará más evidente al referirse a la siguiente memoria descriptiva, reivindicaciones y dibujos adjuntos.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0014] Las formas de realización ilustrativas de la presente solicitud se describen en detalle a continuación con referencia a las siguientes figuras:

35 **[0015]** La Figura 1 es un diagrama de bloques que ilustra un ejemplo de un sistema que incluye un dispositivo de codificación y un dispositivo de decodificación, de acuerdo con

algunos ejemplos;

[0016] La Figura es un diagrama conceptual que ilustra un ejemplo de candidatos de vector de movimiento vecino espacial para un modo de fusión, de acuerdo con algunos ejemplos;

- 5 [0017] La Figura 2B es un diagrama conceptual que ilustra un ejemplo de candidatos de vector de movimiento vecino espacial para un modo de predicción de vector de movimiento avanzado (AMVP), de acuerdo con algunos ejemplos;

[0018] La Figura 3 es un diagrama de flujo que ilustra técnicas para procesar datos de video;

- 10 [0019] La Figura 4 es un diagrama que ilustra aspectos del remuestreo de imágenes de referencia de acuerdo con algunos ejemplos;

[0020] La Figura 5 es un diagrama de flujo que ilustra aspectos de un método para remuestreo de imágenes de referencia de acuerdo con algunos ejemplos;

- 15 [0021] La Figura 6 es un diagrama de flujo que ilustra aspectos del remuestreo de imágenes de referencia de acuerdo con algunos ejemplos;

[0022] La Figura 7 es un diagrama de bloques que ilustra un dispositivo de codificación de video de ejemplo, de acuerdo con algunos ejemplos; y

[0023] La Figura 8 es un diagrama de bloques que ilustra un dispositivo de decodificación de video de ejemplo, de acuerdo con algunos ejemplos.

20

DESCRIPCIÓN DETALLADA

[0024] A continuación, se proporcionan ciertos aspectos y formas de realización de la presente divulgación. Algunos de estos aspectos y formas de realización pueden aplicarse de forma independiente y algunos de ellos pueden aplicarse en combinación, como resultará evidente para los expertos en la técnica. En la siguiente descripción, con fines explicativos, se establecen detalles específicos con el fin de proporcionar una comprensión completa de las formas de realización de la solicitud. Sin embargo, será evidente que se pueden poner en práctica varias formas de realización sin estos detalles específicos. Las figuras y la descripción no pretenden ser restrictivas.

- 25 [0025] La descripción que sigue proporciona formas de realización de ejemplo solamente, y no pretende limitar el alcance, aplicabilidad o configuración de la divulgación. Más bien, la siguiente descripción de las formas de realización de ejemplo proporcionará a los expertos en la técnica una descripción habilitante para implementar una forma de realización de ejemplo. Debe entenderse que se pueden hacer varios cambios en la función y disposición de los elementos sin apartarse del alcance de la
- 30
- 35

aplicación como se establece en las reivindicaciones adjuntas.

[0026] Los dispositivos de codificación de video implementan técnicas de compresión de video para codificar y decodificar datos de video de manera eficiente. Las técnicas de compresión de video pueden incluir la aplicación de diferentes modos de predicción, incluida la predicción espacial (por ejemplo, predicción intracuadros o intrapredicción), predicción temporal (por ejemplo, predicción intercuadros o interpredicción), predicción entre capas (a través de diferentes capas de datos de video y/u otras técnicas de predicción para reducir o eliminar la redundancia inherente a las secuencias de video. Un codificador de video puede dividir cada imagen de una secuencia de video original en regiones rectangulares denominadas bloques de video o unidades de codificación (que se describen con mayor detalle a continuación). Estos bloques se pueden codificar usando un modo de predicción particular.

[0027] Los bloques de video pueden dividirse de una o más formas en uno o más grupos de bloques más pequeños. Los bloques pueden incluir bloques de árbol de codificación, bloques de predicción, bloques de transformación y/u otros bloques adecuados. Las referencias generalmente a un “bloque”, a menos que se especifique lo contrario, pueden referirse a dichos bloques de video (por ejemplo, bloques de árbol de codificación, bloques de codificación, bloques de predicción, bloques de transformación u otros bloques o subbloques apropiados, como entendería un experto en la técnica). Además, cada uno de estos bloques también puede denominarse de manera intercambiable en el presente documento como “unidades” (por ejemplo, unidad de árbol de codificación (CTU), unidad de codificación, unidad de predicción (PU), unidad de transformación (TU) o similares). En algunos casos, una unidad puede indicar una unidad lógica de codificación que está codificada en un flujo de bits, mientras que un bloque puede indicar una parte de la memoria intermedia de tramas de video a la que se dirige un proceso.

[0028] Para los modos de intrapredicción, un codificador de video puede buscar un bloque similar al bloque que se codifica en un cuadro (o imagen) ubicado en otra ubicación temporal, denominado cuadro de referencia o imagen de referencia. El codificador de video puede restringir la búsqueda a cierto desplazamiento espacial desde el bloque por codificar. Una mejor coincidencia se puede encontrar utilizando un vector de movimiento bidimensional (2D) que incluye un componente de desplazamiento horizontal y un componente de desplazamiento vertical. Para los modos de intrapredicción, un codificador de video puede formar el bloque predicho usando técnicas de predicción espacial basadas en datos de bloques vecinos previamente

codificados dentro de la misma imagen.

[0029] El codificador de video puede determinar un error de predicción. Por ejemplo, el error de predicción puede determinarse como la diferencia entre los valores de píxeles en el bloque que se codifica y el bloque predicho. El error de predicción también se puede denominar residual. El codificador de video también puede aplicar una transformada al error de predicción (por ejemplo, una transformada de coseno discreta (DCT) u otra transformada adecuada) para generar coeficientes de transformada. Después de la transformación, el codificador de video puede cuantificar los coeficientes de transformada. Los coeficientes de transformada cuantificados y los vectores de movimiento pueden representarse utilizando elementos de sintaxis y, junto con la información de control, formar una representación codificada de una secuencia de video. En algunos casos, el codificador de video puede entropizar elementos de sintaxis de código, reduciendo así aún más el número de bits necesarios para su representación.

[0030] Un decodificador de video puede construir, utilizando los elementos de sintaxis y la información de control discutida con anterioridad, datos predictivos (por ejemplo, un bloque predictivo) para decodificar un cuadro actual. Por ejemplo, el decodificador de video puede añadir el bloque predicho y el error de predicción comprimido. El decodificador de video puede determinar el error de predicción comprimido al ponderar las funciones de base de transformada utilizando los coeficientes cuantificados. La diferencia entre el cuadro reconstruido y el cuadro original se llama error de reconstrucción.

[0031] En algunos ejemplos, la conversión de frecuencia de muestreo se puede aplicar a datos de video. La conversión de frecuencia de muestreo puede incluir remuestreo, submuestreo, sobremuestreo, interpolación y/u otras técnicas de conversión de frecuencia de muestreo. Por ejemplo, la interpolación puede incluir interpolación constante por partes, interpolación lineal, interpolación polinomial e interpolación spline, entre otras técnicas de interpolación. El remuestreo puede incluir submuestreo, sobremuestreo y/o una combinación de submuestreo y sobremuestreo, o la obtención de nuevas muestras a una frecuencia de muestreo diferente.

[0032] En ciertos sistemas, la predicción de movimiento se puede mejorar con una técnica de fusión de bloques de interpredicción para simplificar la señalización de datos de movimiento por bloques. Las tareas de interpolación se utilizan en dicho contexto debido a que el desplazamiento de objetos entre imágenes es independiente de la cuadrícula de píxeles de las imágenes. La precisión fraccionaria (por ejemplo, la estimación de la ubicación entre píxeles en las imágenes) se puede utilizar para capturar

el movimiento con mayor precisión. Esta precisión fraccionaria puede permitir la inferencia de datos de movimiento a partir de bloques previamente decodificados. Para muestras de imágenes de referencia de interpolación y fracción, la alta precisión puede mejorar el filtrado. Los sistemas de ejemplo pueden admitir vectores de movimiento con una precisión de un cuarto de píxel, una precisión de un octavo de píxel, una precisión de un dieciseisavo de píxel u otra precisión similar, según el sistema. En algunos ejemplos, los componentes de luminancia y crominancia pueden tener una precisión fraccionaria diferente en el mismo sistema. Los filtros de interpolación se pueden utilizar para procesar imágenes en este contexto. Además, se pueden usar múltiples filtros de interpolación diferentes en diferentes situaciones (por ejemplo, según los valores del vector de movimiento). La selección entre diferentes filtros de interpolación que están disponibles se denomina en el presente documento conmutación de filtro de interpolación.

[0033] El remuestreo se refiere a tomar valores de píxeles de una imagen y “remuestrear” los datos de los valores de píxeles para generar nuevos valores de píxeles para una nueva imagen. Como se describió con anterioridad, el remuestreo puede incluir un submuestreo, en donde los valores de píxeles de una imagen de mayor resolución se utilizan para generar una imagen a una resolución más baja, así como el sobremuestreo, en donde los valores de píxeles de una imagen de resolución más baja se utilizan para generar una imagen a una resolución más alta. En el contexto de la predicción de movimiento descrito con anterioridad, las imágenes de referencia y las imágenes actuales con diferentes tamaños (por ejemplo, resoluciones) se pueden usar para generar vectores de movimiento y realizar predicciones de movimiento. El remuestreo se puede utilizar en tales situaciones para gestionar la diferencia de escala entre la imagen de referencia y la imagen actual. Además, cuando las imágenes de referencia y actuales no son del mismo tamaño, un filtro puede indicarse mediante un índice de filtro que es adecuado para el filtrado de interpolación sin remuestreo, pero no para el filtrado de interpolación con remuestreo de la imagen de referencia. En algunos sistemas de procesamiento de imágenes, por ejemplo, un filtro de suavizado utilizado para la interpolación puede generar resultados deficientes cuando se utiliza para el remuestreo de imágenes de referencia. Los ejemplos descritos en el presente documento mejoran el funcionamiento de los dispositivos de decodificación y codificación identificando valores de filtro apropiados y generando resultados de codificación y decodificación mejorados durante las operaciones de los dispositivos de codificación y decodificación. En un ejemplo, esta mejora se implementa identificando

la información de índice de filtro para un bloque actual de la imagen actual y determinando que un primer valor de tamaño de imagen de la imagen actual y un segundo valor de tamaño de imagen de la imagen de referencia son diferentes (por ejemplo, se utiliza ese remuestreo de imágenes de referencia). En este contexto, el ejemplo realiza un proceso de remuestreo usando un índice de filtro predeterminado en lugar de un índice de filtro actual identificado por la información del índice de filtro (por ejemplo, sin usar el índice de filtro actual). A continuación, se describen aspectos adicionales de varios ejemplos.

[0034] Las técnicas descritas en este documento se pueden aplicar a cualquiera de los códecs de video existentes (por ejemplo, codificación de video de alta eficiencia (HEVC), codificación de video avanzada (AVC) u otro códec de video existente adecuado) y/o puede ser una herramienta de codificación eficiente para cualquier estándar de codificación de video que se esté desarrollando y/o estándares de codificación de video futuros, como, por ejemplo, codificación de video versátil (VVC), el modelo de exploración conjunta (JEM) y/u otros estándares de codificación de video en desarrollo o por desarrollar.

[0035] La Figura 1 es un diagrama de bloques que ilustra un ejemplo de un sistema que incluye un dispositivo de codificación 104 y un dispositivo de decodificación 112. El dispositivo de codificación 104 puede ser parte de un dispositivo de origen, y el dispositivo de decodificación 112 puede ser parte de un dispositivo receptor. El dispositivo de origen y/o el dispositivo receptor pueden incluir un dispositivo electrónico, como un teléfono móvil o fijo (por ejemplo, un teléfono inteligente, un teléfono celular o similar), una computadora de escritorio, una computadora laptop o notebook, una tableta, un decodificador, un televisor, una cámara, un dispositivo de visualización, un reproductor de medios digitales, una consola de videojuegos, un dispositivo de transmisión de video, una cámara de protocolo de Internet (IP), o cualquier otro dispositivo electrónico adecuado. En algunos ejemplos, el dispositivo de origen y el dispositivo receptor pueden incluir uno o más transceptores inalámbricos para comunicaciones inalámbricas. Las técnicas de codificación descritas en este documento son aplicables a la codificación de video en varias aplicaciones multimedia, incluidas transmisiones de video en tiempo real (por ejemplo, a través de Internet), transmisiones o difusiones de televisión, codificación de video digital para su almacenamiento en un medio de almacenamiento de datos, decodificación de video digital almacenado en un medio de almacenamiento de datos u otras aplicaciones. En algunos ejemplos, el sistema 100 puede admitir la transmisión de video unidireccional o bidireccional para

admitir aplicaciones tales como videoconferencia, transmisión continua de video, reproducción de video, transmisión de video, juegos y/o telefonía de video.

[0036] El dispositivo de codificación 104 (o codificador) se puede utilizar para codificar datos de video utilizando un protocolo o estándar de codificación de video para generar un flujo de bits de video codificado. Ejemplos de 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, ITU-T H.264 (también conocido como ISO/IEC MPEG-4 AVC), incluidas sus extensiones de codificación de video escalable (SVC) y codificación de video de vista múltiple (MVC), y codificación de video de alta eficiencia (HEVC) o ITU-T H.265. Existen varias extensiones para HEVC que se ocupan de la codificación de video multicapa, incluidas las extensiones de codificación de contenido de pantalla y rango, la codificación de video 3D (3D-HEVC) y las extensiones de vista múltiple (MV-HEVC) y la extensión escalable (SHVC). El HEVC y sus extensiones han sido desarrollados por el Equipo de colaboración conjunta sobre codificación de video (JCT-VC), así como el Equipo de colaboración conjunta sobre el desarrollo de extensiones de codificación de video 3D (JCT-3V) del Grupo de expertos en codificación de video del UIT-T (VCEG) y Grupo de expertos en imágenes en movimiento ISO/IEC (MPEG). MPEG y ITU-T VCEG también han formado un equipo de video de exploración conjunta (JVET) para explorar nuevas herramientas de codificación para la próxima generación de estándares de codificación de video. El software de referencia se llama JEM (modelo de exploración conjunta). Un nuevo estándar de codificación de video que está desarrollando JVET se llama Codificación de video versátil (Versatile Video Coding, VVC).

[0037] HEVC fue finalizado por el Equipo de colaboración conjunta sobre codificación de video (JCT-VC) del Grupo de expertos en codificación de video del UIT-T (VCEG) y el Grupo de expertos en imágenes en movimiento de ISO/IEC (MPEG) en 2013. El equipo conjunto de expertos en video (JVET), un equipo colaborativo formado por MPEG y VCEG de la Comisión de Estudio 16 del UIT-T está trabajando en el nuevo estándar de codificación de video VVC. Un objetivo de VVC es proporcionar una mejora significativa en el rendimiento de compresión sobre el estándar HEVC existente, ayudando en la implementación de servicios de video de mayor calidad y aplicaciones emergentes (por ejemplo, como multimedia inmersiva omnidireccional de 360°, video de alto rango dinámico (HDR), entre otros).

[0038] Muchas formas de realización descritas en el presente documento proporcionan ejemplos que utilizan el modelo JEM, VVC, el estándar HEVC y/o extensiones del

mismo. Sin embargo, como se señaló con anterioridad, las técnicas y sistemas descritos en este documento también pueden ser aplicables a otros estándares de codificación, como AVC, MPEG, JPEG (u otro estándar de codificación para imágenes fijas), extensiones de los mismos u otros estándares de codificación adecuados ya disponibles o aún no disponible o desarrollado. Por consiguiente, aunque las técnicas y sistemas descritos en el presente documento pueden describirse con referencia a un estándar de codificación de video particular, un experto en la técnica apreciará que la descripción no debe interpretarse como aplicable únicamente a ese estándar particular.

[0039] Haciendo referencia a la Figura 1, una fuente de video 102 puede proporcionar los datos de video al dispositivo de codificación 104. La fuente de video 102 puede ser parte del dispositivo de origen, o puede ser parte de un dispositivo que no sea el dispositivo de origen. La fuente de video 102 puede incluir un dispositivo de captura de video (por ejemplo, una cámara de video, un teléfono con cámara, un teléfono de video o similar), un archivo de video que contiene video almacenado, un servidor de video o proveedor de contenido que proporciona datos de video, una interfaz de alimentación de video que recibe video de un servidor de video o proveedor de contenido, un sistema de gráficos de computadora para generar datos de video de gráficos de computadora, una combinación de tales fuentes o cualquier otra fuente de video adecuada.

[0040] Los datos de video de la fuente 102 de video pueden incluir una o más imágenes o cuadros de entrada. Una imagen o cuadro es una imagen fija que, en algunos casos, forma parte de un video. En algunos ejemplos, los datos de la fuente 102 de video pueden ser una imagen fija que no forma parte de un video. El motor codificador 106 (o codificador) del dispositivo codificador 104 codifica los datos de video para generar un flujo de bits de video codificado. En algunos ejemplos, un flujo de bits de video codificado (o “flujo de bits de video” o “flujo de bits”) es una serie de una o más secuencias de video codificadas. Una secuencia de video codificada (CVS) incluye una serie de unidades de acceso (AU) que comienzan con una AU que tiene una imagen de punto de acceso aleatorio en la capa base y con ciertas propiedades hasta y sin incluir una próxima AU que tenga una imagen de punto de acceso aleatorio en la capa base y con ciertas propiedades. Por ejemplo, ciertas propiedades de una imagen de punto de acceso aleatorio que inicia una CVS pueden incluir un indicador RASL (por ejemplo, NoRaslOutputFlag) igual a 1. De lo contrario, una imagen de punto de acceso aleatorio (con una marca RASL igual a 0) no inicia una CVS. Una unidad de acceso (AU) incluye una o más imágenes codificadas e información de control correspondiente a las imágenes codificadas que comparten el mismo tiempo de salida. Los segmentos de

- imágenes codificadas se encapsulan en el nivel de flujo de bits en unidades de datos llamadas unidades de capa de abstracción de red (NAL). Por ejemplo, un flujo de bits de video HEVC puede incluir uno o más CVS incluyendo unidades NAL. Cada una de las unidades NAL tiene un encabezado de unidad NAL. En un ejemplo, el encabezado es de un byte para H.264/AVC (excepto para extensiones multicapa) y de dos bytes para HEVC. Los elementos de sintaxis en el encabezado de la unidad NAL toman los bits designados y, por lo tanto, son visibles para todo tipo de sistemas y capas de transporte, como Transport Stream, Real-time Transport (RTP) Protocol, File Format, entre otros.
- 5
- 10 **[0041]** Existen dos clases de unidades NAL en el estándar HEVC, incluidas las unidades NAL de la capa de codificación de video (VCL) y las unidades NAL que no son VCL. Una unidad VCL NAL incluye un segmento o porción de segmento (descrito a continuación) de datos de imagen codificados, y una unidad NAL no VCL incluye información de control que se relaciona con una o más imágenes codificadas. En algunos casos, una unidad NAL puede denominarse paquete. Una HEVC AU incluye unidades VCL NAL que contienen datos de imagen codificados y unidades NAL no VCL (si las hay) correspondientes a los datos de imagen codificados.
- 15
- [0042]** Las unidades NAL pueden contener una secuencia de bits que forman una representación codificada de los datos de video (por ejemplo, un flujo de bits de video codificado, una CVS de un flujo de bits, o similares), como representaciones codificadas de imágenes en un video. El motor de codificación 106 genera representaciones codificadas de imágenes dividiendo cada imagen en múltiples porciones. Un segmento es independiente de otros segmentos, de modo que la información del segmento se codifica sin depender de los datos de otros segmentos de la misma imagen. Un segmento incluye una o más porciones de segmento que incluyen una porción de segmento independiente y, si está presente, una o más porciones de segmento dependientes que dependen de porciones de segmento anteriores. A continuación, los segmentos se dividen luego en bloques de árbol de codificación (CTB) de muestras de luminancia y muestras de crominancia. Un CTB de muestras de luminancia y uno o más
- 20
- 25
- 30 CTB de muestras de crominancia, junto con la sintaxis de las muestras, se denominan unidad de árbol de codificación (CTU). Una CTU es la unidad de procesamiento básica para la codificación HEVC. Una CTU se puede dividir en varias unidades de codificación (CU) de distintos tamaños. Una CU contiene matrices de muestras de luminancia y crominancia que se denominan bloques de codificación (CB).
- 35 **[0043]** Los CB de luminancia y crominancia pueden dividirse adicionalmente en bloques

de predicción (PB). Un PB es un bloque de muestras del componente de luminancia o un componente de crominancia que utiliza los mismos parámetros de movimiento para interpredicción o predicción de copias intrabloques (cuando está disponible o habilitado para su uso). El PB de luminancia y uno o más PB de crominancia, junto con la sintaxis asociada, forman una unidad de predicción (PU). Para la interpredicción, un conjunto de parámetros de movimiento (por ejemplo, uno o más vectores de movimiento, índices de referencia o similares) se señala en el flujo de bits para cada PU y se utiliza para la interpredicción de PB de luminancia y uno o más PB de crominancia. Los parámetros de movimiento también pueden denominarse información de movimiento. Un CB también se puede dividir en uno o más bloques de transformación (TB). Un TB representa un bloque cuadrado de muestras de un componente de color en donde se aplica la misma transformada bidimensional para codificar una señal residual de predicción. Una unidad de transformación (TU) representa los TB de las muestras de luminancia y crominancia, y los elementos de sintaxis correspondientes.

5 [0044] Un tamaño de una CU corresponde a un tamaño del modo de codificación y puede ser de forma cuadrada. Por ejemplo, un tamaño de CU puede ser de 8 x 8 muestras, 16 x 16 muestras, 32 x 32 muestras, 64 x 64 muestras, o cualquier otro tamaño apropiado hasta el tamaño de la CTU correspondiente. La frase "N x N" se usa aquí para referirse a las dimensiones en píxeles de un bloque de video en términos de

10 dimensiones verticales y horizontales (por ejemplo, 8 píxeles x 8 píxeles). Los píxeles en un bloque se pueden organizar en filas y columnas. En algunos ejemplos, los bloques pueden no tener el mismo número de píxeles en una dirección horizontal que en una dirección vertical. Los datos de sintaxis asociados con una CU pueden describir, por ejemplo, la partición de la CU en una o más PU. Los modos de partición pueden diferir entre si la CU está codificada en modo de intrapredicción o en modo de interpredicción. Las PU se pueden dividir para que no tengan forma cuadrada. Los datos de sintaxis asociados con una CU también pueden describir, por ejemplo, la partición de la CU en una o más TU según una CTU. Una TU puede tener forma cuadrada o no cuadrada.

15 [0045] Según el estándar HEVC, las transformaciones pueden realizarse usando unidades de transformación (TU). Las TU pueden variar para diferentes CU. Las TU pueden dimensionarse en función del tamaño de las PU dentro de una CU determinada. Las TU pueden ser del mismo tamaño o más pequeñas que las PU. En algunos ejemplos, las muestras residuales correspondientes a un CU se pueden subdividir en unidades más pequeñas utilizando una estructura quadtree conocida como árbol

20 cuádruple residual (RQT). Los nodos foliares de la RQT pueden corresponder a TU. Los

25

30

35

valores de diferencia de píxeles asociados con las TU pueden transformarse para producir coeficientes de transformada. Los coeficientes de transformada pueden ser cuantificados luego por el motor codificador 106.

5 **[0046]** Una vez que las imágenes de los datos de video se dividen en CU, el motor codificador 106 predice cada PU utilizando un modo de predicción. La unidad de predicción o el bloque de predicción se resta luego de los datos de video originales para obtener residuos (descritos a continuación). Para cada CU, se puede indicar un modo de predicción dentro del flujo de bits utilizando datos de sintaxis. Un modo de predicción puede incluir intrapredicción (o predicción intraimagen) o interpredicción (o predicción interimagen). La intrapredicción utiliza la correlación entre muestras espacialmente vecinas dentro de una imagen. Por ejemplo, usando intrapredicción, cada PU se predice a partir de datos de imágenes vecinas en la misma imagen usando, por ejemplo, predicción DC para encontrar un valor promedio para el PU, predicción plana para ajustar una superficie plana al PU, predicción de dirección para la extrapolación de datos vecinos, o cualquier otro tipo de predicción adecuado. La interpredicción utiliza la correlación temporal entre imágenes para derivar una predicción con compensación de movimiento para un bloque de muestras de imágenes. Por ejemplo, usando la interpredicción, cada PU se predice usando la predicción de compensación de movimiento a partir de datos de imagen en una o más imágenes de referencia (antes o después de la imagen actual en orden de salida). La decisión de codificar un área de imagen usando predicción interimágenes o intraimágenes puede hacerse, por ejemplo, a nivel de CU.

10

15

20

[0047] En algunos ejemplos, a uno o más segmentos de una imagen se les asigna un tipo de segmento. Los tipos de segmento incluyen un segmento I, un segmento P y un segmento B. Un segmento I (intracuadros, decodificable de forma independiente) es un segmento de una imagen que solo se codifica mediante intrapredicción y, por lo tanto, se puede decodificar de forma independiente, ya que el segmento I solo requiere los datos dentro del cuadro para predecir cualquier unidad de predicción o bloque de predicción del segmento. Un segmento P (cuadros predichos unidireccionales) es un segmento de una imagen que puede codificarse con intrapredicción y con interpredicción unidireccional. Cada unidad de predicción o bloque de predicción dentro de un segmento P se codifica con intrapredicción o interpredicción. Cuando se aplica la interpredicción, la unidad de predicción o el bloque de predicción solo se predice mediante una imagen de referencia y, por lo tanto, las muestras de referencia son solo de una región de referencia de un cuadro. Un segmento B (marcos predictivos

25

30

35

bidireccionales) es un segmento de una imagen que puede codificarse con intrapredicción y con interpredicción (por ejemplo, ya sea bipredicción o unipredicción). Una unidad de predicción o un bloque de predicción de un segmento B puede predecirse bidireccionalmente a partir de dos imágenes de referencia, en donde cada imagen

5 aporta una región de referencia y los conjuntos de muestras de las dos regiones de referencia se ponderan (por ejemplo, con pesos iguales o con pesos diferentes) para producir la señal de predicción del bloque predicho bidireccional. Como se explicó con anterioridad, los segmentos de una imagen se codifican de forma independiente. En algunos casos, una imagen se puede codificar como un solo segmento.

10 **[0048]** Como se señaló con anterioridad, la predicción intraimagen utiliza la correlación entre muestras vecinas espacialmente dentro de una imagen. La predicción interimágenes utiliza la correlación temporal entre imágenes para obtener una predicción compensada por movimiento para un bloque de muestras de imágenes. Usando un modelo de movimiento traslacional, la posición de un bloque en una imagen

15 previamente decodificada (una imagen de referencia) se indica mediante un vector de movimiento $(\Delta x, \Delta y)$, en donde Δx especifica el desplazamiento horizontal y Δy especifica el desplazamiento vertical del bloque de referencia relativo a la posición del bloque actual. En algunos casos, un vector de movimiento $(\Delta x, \Delta y)$ puede tener una precisión de muestra entera (también conocida como precisión entera), en cuyo caso el

20 vector de movimiento apunta a la cuadrícula de "integer-pel" (o cuadrícula de muestreo de píxeles enteros) del marco de referencia. En algunos casos, un vector de movimiento $(\Delta x, \Delta y)$ puede tener una precisión de muestra fraccionaria (también conocida como precisión "fractional-pel" o precisión no entera) para capturar con mayor precisión el movimiento del objeto subyacente, sin limitarse a la cuadrícula de números enteros del

25 marco de referencia. La precisión de los vectores de movimiento puede expresarse por el nivel de cuantificación de los vectores de movimiento. Por ejemplo, el nivel de cuantificación puede ser la precisión entera (por ejemplo, 1 píxel) o la precisión "fractional-pel" (por ejemplo, $\frac{1}{4}$ píxel, $\frac{1}{2}$ píxel u otro valor de subpíxel). La interpolación se aplica en imágenes de referencia para derivar la señal de predicción cuando el vector

30 de movimiento correspondiente tiene una precisión de muestra fraccionaria. Por ejemplo, las muestras disponibles en posiciones enteras se pueden filtrar (por ejemplo, usando uno o más filtros de interpolación) para estimar valores en posiciones fraccionarias. La imagen de referencia previamente decodificada se indica mediante un índice de referencia (refIdx) a una lista de imágenes de referencia. Los vectores de

35 movimiento y los índices de referencia pueden denominarse parámetros de movimiento.

Se pueden realizar dos tipos de predicción interimágenes, incluida la unipredicción y la bipredicción.

[0049] Con la interpredicción usando la bipredicción, se utilizan dos conjuntos de parámetros de movimiento ($\Delta x_0, \Delta y_0, refIdx_0$ y $\Delta x_1, \Delta y_1, refIdx_1$) para generar dos predicciones compensadas de movimiento (de la misma imagen de referencia o posiblemente de diferentes imágenes de referencia). Por ejemplo, con la bipredicción, cada bloque de predicción usa dos señales de predicción compensadas de movimiento y genera unidades de predicción B. Las dos predicciones compensadas de movimiento se combinan luego para obtener la predicción compensada de movimiento final. Por ejemplo, las dos predicciones compensadas de movimiento se pueden combinar haciendo un promedio. En otro ejemplo, se puede utilizar la predicción ponderada, en cuyo caso se pueden aplicar diferentes pesos a cada predicción compensada por movimiento. Las imágenes de referencia que se pueden usar en bipredicción se almacenan en dos listas separadas, denotadas como lista 0 y lista 1. Los parámetros de movimiento se pueden derivar en el codificador utilizando un proceso de estimación de movimiento.

[0050] Con la interpredicción utilizando unipredicción, se utiliza un conjunto de parámetros de movimiento ($\Delta x_0, \Delta y_0, refIdx_0$) para generar una predicción compensada de movimiento a partir de una imagen de referencia. Por ejemplo, con la unipredicción, cada bloque de predicción utiliza como máximo una señal de predicción compensada por movimiento y genera unidades de predicción P.

[0051] Una PU puede incluir los datos (por ejemplo, parámetros de movimiento u otros datos adecuados) relacionados con el proceso de predicción. Por ejemplo, cuando la PU se codifica utilizando la intrapredicción, la PU puede incluir datos que describen un modo de intrapredicción para la PU. Como otro ejemplo, cuando la PU se codifica utilizando interpredicción, la PU puede incluir datos que definen un vector de movimiento para la PU. Los datos que definen el vector de movimiento para un PU pueden describir, por ejemplo, un componente horizontal del vector de movimiento (Δx), un componente vertical del vector de movimiento (Δy), una resolución para el vector de movimiento (por ejemplo, precisión entera, una precisión de cuarto de píxel o una precisión de octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento, un índice de referencia, una lista de imágenes de referencia (por ejemplo, Lista 0, Lista 1 o Lista C) para el vector de movimiento, o cualquier combinación de ellos.

[0052] El dispositivo de codificación 104 puede entonces realizar transformación y cuantificación. Por ejemplo, después de la predicción, el motor 106 del codificador puede

calcular valores residuales correspondientes a la PU. Los valores residuales pueden comprender valores de diferencia de píxeles entre el bloque actual de píxeles que se codifican (la PU) y el bloque de predicción utilizado para predecir el bloque actual (por ejemplo, la versión predicha del bloque actual). Por ejemplo, después de generar un

5 bloque de predicción (por ejemplo, usando interpredicción o intrapredicción), el motor codificador 106 puede generar un bloque residual restando el bloque de predicción producido por una unidad de predicción del bloque actual. El bloque residual incluye un conjunto de valores de diferencia de píxeles que cuantifican las diferencias entre los valores de píxeles del bloque actual y los valores de píxeles del bloque de predicción.

10 En algunos ejemplos, el bloque residual puede representarse en un formato de bloque bidimensional (por ejemplo, una matriz bidimensional o una matriz de valores de píxeles). En tales ejemplos, el bloque residual es una representación bidimensional de los valores de píxel.

[0053] Cualquier dato residual que pueda quedar después de realizar la predicción se

15 transforma usando una transformación de bloque, que puede basarse en una transformada discreta de coseno, una transformada sinusoidal discreta, una transformada entera, una transformada de ondas pequeñas, otra función de transformación adecuada, o cualquier combinación de ellas. En algunos casos, una o más transformaciones de bloque (por ejemplo, tamaños 32 x 32, 16 x 16, 8 x 8, 4 x 4 u

20 otro tamaño adecuado) pueden aplicarse a los datos residuales en cada CU. En algunas formas de realización, se puede usar una TU para los procesos de transformación y cuantificación implementados por el motor codificador 106. Una CU dada que tiene una o más PU también puede incluir una o más TU. Como se describe con más detalle a continuación, los valores residuales pueden transformarse en coeficientes de

25 transformada usando las transformaciones de bloque, y luego pueden cuantificarse y escanearse usando TU para producir coeficientes de transformada serializados para la codificación de entropía.

[0054] En algunas formas de realización después de la codificación intrapredictiva o interpredictiva usando PU de una CU, el motor codificador 106 puede calcular datos

30 residuales para las TU de la CU. Las PU pueden comprender datos de píxeles en el dominio espacial (o dominio de píxeles). Las TU pueden comprender coeficientes en el dominio de transformada después de la aplicación de una transformada de bloque. Como se señaló con anterioridad, los datos residuales pueden corresponder a valores de diferencia de píxeles entre píxeles de la imagen no codificada y valores de predicción

35 correspondientes a las PU. El motor codificador 106 puede formar las TU que incluyen

los datos residuales para la CU, y luego puede transformar las TU para producir coeficientes de transformada para la CU.

[0055] El motor codificador 106 puede realizar la cuantificación de los coeficientes de transformada. La cuantificación proporciona una compresión adicional al cuantificar los coeficientes de transformada para reducir la cantidad de datos utilizados para representar los coeficientes. Por ejemplo, la cuantificación puede reducir la profundidad de bits asociada con algunos o todos los coeficientes. En un ejemplo, un coeficiente con un valor de n bits puede redondearse a un valor de m bits durante la cuantificación, siendo n mayor que m .

[0056] Una vez que se realiza la cuantificación, el flujo de bits de video codificado incluye coeficientes de transformada cuantificados, información de predicción (por ejemplo, modos de predicción, vectores de movimiento, vectores de bloque o similares), información de partición y cualquier otro dato adecuado, tal como otros datos de sintaxis. Los diferentes elementos del flujo de bits de video codificado pueden ser codificados luego por entropía por el motor codificador 106. En algunos ejemplos, el motor codificador 106 puede utilizar un orden de escaneo predefinido para escanear los coeficientes de transformada cuantificados para producir un vector serializado que puede codificarse por entropía. En algunos ejemplos, el motor codificador 106 puede realizar un escaneo adaptativo. Después de escanear los coeficientes de transformada cuantificados para formar un vector (por ejemplo, un vector unidimensional), el motor codificador 106 puede codificar por entropía el vector. Por ejemplo, el motor 106 del codificador puede usar codificación adaptativa de longitud variable de contexto, codificación aritmética binaria adaptativa de contexto, codificación aritmética binaria adaptativa de contexto basada en sintaxis, codificación de entropía de partición de intervalo de probabilidad u otra técnica de codificación de entropía adecuada.

[0057] La salida 110 del dispositivo de codificación 104 puede enviar las unidades NAL que componen los datos de flujo de bits de video codificados a través del enlace de comunicaciones 120 al dispositivo de decodificación 112 del dispositivo receptor. La entrada 114 del dispositivo de decodificación 112 puede recibir las unidades NAL. El enlace de comunicaciones 120 puede incluir un canal proporcionado por una red inalámbrica, una red cableada o una combinación de una red cableada e inalámbrica. Una red inalámbrica puede incluir cualquier interfaz inalámbrica o combinación de interfaces inalámbricas y puede incluir cualquier red inalámbrica adecuada (por ejemplo, Internet u otra red de área amplia, una red basada en paquetes, WiFi™, radiofrecuencia (RF), UWB, WiFi-Direct, celular, evolución a largo plazo (LTE), WiMax™, o similares).

Una red cableada puede incluir cualquier interfaz cableada (por ejemplo, fibra, ethernet, ethernet powerline, ethernet a través de cable coaxial, línea de señal digital (DSL), o similares). Las redes cableadas y/o inalámbricas pueden implementarse utilizando diversos equipos, tales como estaciones base, enrutadores, puntos de acceso, puentes,

5 puertas de enlace, conmutadores, o similares. Los datos de flujo de bits de video codificados pueden modularse de acuerdo con un estándar de comunicación, tal como un protocolo de comunicación inalámbrico, y transmitirse al dispositivo receptor.

[0058] En algunos ejemplos, el dispositivo de codificación 104 puede almacenar datos codificados de flujo de bits de video en el almacenamiento 108. La salida 110 puede

10 recuperar los datos codificados de flujo de bits de video del motor codificador 106 o del almacenamiento 108. El almacenamiento 108 puede incluir cualquiera de una variedad de medios de almacenamiento de datos distribuidos o localmente accedidos. Por ejemplo, el almacenamiento 108 puede incluir un disco duro, un disco de almacenamiento, memoria flash, memoria volátil o no volátil, o cualquier otro medio de

15 almacenamiento digital adecuado para almacenar datos de video codificados.

[0059] La entrada 114 del dispositivo de decodificación 112 recibe los datos de flujo de bits de video codificados y puede proporcionar los datos de flujo de bits de video al motor de decodificador 116, o al almacenamiento 118 para uso posterior por el motor de decodificador 116. El motor de decodificador 116 puede decodificar los datos de flujo de

20 bits de video codificados mediante decodificación de entropía (por ejemplo, utilizando un decodificador de entropía) y extrayendo los elementos de una o más secuencias de video codificadas que forman los datos de video codificados. El motor decodificador 116 puede reescalar luego y realizar una transformada inversa en los datos de flujo de bits de video codificados. Los datos residuales luego se pasan a una etapa de predicción

25 del motor decodificador 116. El motor decodificador 116 luego predice un bloque de píxeles (por ejemplo, una PU). En algunos ejemplos, la predicción se agrega a la salida de la transformada inversa (los datos residuales).

[0060] El dispositivo de decodificación 112 puede emitir el video decodificado a un dispositivo de destino de video 122, que puede incluir una pantalla u otro dispositivo de

30 salida para mostrar los datos de video decodificados a un consumidor del contenido. En algunos aspectos, el dispositivo de destino de video 122 puede ser parte del dispositivo receptor que incluye el dispositivo de decodificación 112. En algunos aspectos, el dispositivo de destino de video 122 puede ser parte de un dispositivo separado que no sea el dispositivo receptor.

35 **[0061]** En algunas formas de realización, el dispositivo de codificación de video 104 y/o

el dispositivo de decodificación de video 112 pueden integrarse con un dispositivo de codificación de audio y un dispositivo de decodificación de audio, respectivamente. El dispositivo de codificación de video 104 y/o el dispositivo de decodificación de video 112 también pueden incluir otro hardware o software que sea necesario para implementar

5 las técnicas de codificación descritas con anterioridad, como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables en campo (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. El dispositivo de codificación de video 104 y el dispositivo de decodificación de

10 video 112 pueden integrarse como parte de un codificador/decodificador combinado (códec) en un dispositivo respectivo. Un ejemplo de detalles específicos del dispositivo de codificación 104 se describe a continuación con referencia a la Figura 7. Un ejemplo de detalles específicos del dispositivo de decodificación 112 se describe a continuación con referencia a la Figura 8.

15 **[0062]** Las extensiones del estándar HEVC incluyen la extensión de codificación de video de vista múltiple, denominada MV-HEVC, y la extensión de codificación de video escalable, denominada SHVC. Las extensiones MV-HEVC y SHVC comparten el concepto de codificación en capas, con diferentes capas incluidas en el flujo de bits de video codificado. Cada capa de una secuencia de video codificada se direcciona

20 mediante un identificador de capa (ID) único. Un ID de capa puede estar presente en un encabezado de una unidad NAL para identificar una capa con la que está asociada la unidad NAL. En MV-HEVC, diferentes capas pueden representar diferentes vistas de la misma escena en el flujo de bits de video. En SHVC, se proporcionan diferentes capas escalables que representan el flujo de bits de video en diferentes resoluciones

25 espaciales (o resolución de imagen) o en diferentes fidelidades de reconstrucción. Las capas escalables pueden incluir una capa base (con ID de capa = 0) y una o más capas de mejora (con ID de capa = 1, 2,... n). La capa base puede ajustarse a un perfil de la primera versión de HEVC y representa la capa más baja disponible en un flujo de bits. Las capas de mejora tienen mayor resolución espacial, resolución temporal o tasa de

30 cuadros y/o fidelidad (o calidad) de reconstrucción en comparación con la capa base. Las capas de mejora están organizadas jerárquicamente y pueden (o no) depender de capas inferiores. En algunos ejemplos, las diferentes capas se pueden codificar utilizando un único códec estándar (por ejemplo, todas las capas se codifican utilizando HEVC, SHVC u otro estándar de codificación). En algunos ejemplos, se pueden codificar

35 diferentes capas utilizando un códec multiestándar. Por ejemplo, una capa base puede

codificarse utilizando AVC, mientras que una o más capas de mejora pueden codificarse utilizando extensiones SHVC y/o MV-HEVC del estándar HEVC.

[0063] En general, una capa incluye un conjunto de unidades VCL NAL y un conjunto correspondiente de unidades no VCL NAL. A las unidades NAL se les asigna un valor de ID de capa particular. Las capas pueden ser jerárquicas en el sentido de que una capa puede depender de una capa inferior. Un conjunto de capas se refiere a un conjunto de capas representadas dentro de un flujo de bits que son autónomas, lo que significa que las capas dentro de un conjunto de capas pueden depender de otras capas en el conjunto de capas en el proceso de decodificación, pero no dependen de ninguna otra capa para decodificación. Por consiguiente, las capas de un conjunto de capas pueden formar un flujo de bits independiente que puede representar contenido de video. El conjunto de capas de un conjunto de capas puede obtenerse de otro flujo de bits mediante la operación de un proceso de extracción de subflujo de bits. Un conjunto de capas puede corresponder al conjunto de capas que se va a decodificar cuando un decodificador desea operar de acuerdo con ciertos parámetros.

[0064] Como se describió previamente, un flujo de bits HEVC incluye un grupo de unidades NAL, que incluyen unidades NCL VCL y unidades NAL no VCL. Las unidades VCL NAL incluyen datos de imagen codificados que forman un flujo de bits de video codificado. Por ejemplo, una secuencia de bits que forman el flujo de bits de video codificado está presente en las unidades VCL NAL. Las unidades NAL no VCL pueden contener conjuntos de parámetros con información de alto nivel relacionada con el flujo de bits de video codificado, además de otra información. Por ejemplo, un conjunto de parámetros puede incluir un conjunto de parámetros de video (VPS), un conjunto de parámetros de secuencia (SPS) y un conjunto de parámetros de imagen (PPS). Los ejemplos de objetivos de los conjuntos de parámetros incluyen la eficiencia de la tasa de bits, la resistencia a los errores y el suministro de interfaces de capa de sistemas. Cada segmento hace referencia a un único PPS, SPS y VPS activo para acceder a la información que el dispositivo de decodificación 112 puede usar para decodificar el segmento. Se puede codificar un identificador (ID) para cada conjunto de parámetros, que incluye una ID de VPS, una ID de SPS y una ID de PPS. Un SPS incluye una ID de SPS y una ID de VPS. Un PPS incluye una ID de PPS y una ID de SPS. Cada encabezado de segmento incluye una ID de PPS. Usando las ID, se pueden identificar conjuntos de parámetros activos para un segmento dado.

[0065] Un PPS incluye información que se aplica a todos los segmentos en una imagen dada. Debido a esto, todos los segmentos en una imagen se refieren al mismo PPS. Los

segmentos en diferentes imágenes también pueden referirse al mismo PPS. Un SPS incluye información que se aplica a todas las imágenes en una misma secuencia de video codificada (CVS) o flujo de bits. Como se describió con anterioridad, una secuencia de video codificada es una serie de unidades de acceso (AU) que comienza con una imagen de punto de acceso aleatorio (por ejemplo, una imagen de referencia de decodificación instantánea (IDR) o una imagen de acceso de enlace roto (BLA) u otra imagen de punto de acceso aleatorio apropiada) en la capa base y con ciertas propiedades (descritas con anterioridad) hasta, sin incluir, una siguiente AU que tiene una imagen de punto de acceso aleatorio en la capa base y con ciertas propiedades (o el final del flujo de bits). La información en un SPS puede no cambiar de una imagen a otra dentro de una secuencia de video codificada. Las imágenes en una secuencia de video codificada pueden usar el mismo SPS. El VPS incluye información que se aplica a todas las capas dentro de una secuencia de video codificada o flujo de bits. El VPS incluye una estructura de sintaxis con elementos de sintaxis que se aplican a secuencias de video codificadas completas. En algunas formas de realización, el VPS, SPS o PPS pueden transmitirse en banda con el flujo de bits codificado. En algunas formas de realización, el VPS, SPS o PPS pueden transmitirse fuera de banda en una transmisión separada que las unidades NAL que contienen datos de video codificados.

[0066] Un flujo de bits de video también puede incluir mensajes de información de mejora suplementaria (SEI). Por ejemplo, una unidad SEI NAL puede ser parte del flujo de bits de video. En algunos casos, un mensaje SEI puede contener información que no es necesaria para el proceso de decodificación. Por ejemplo, la información en un mensaje SEI puede no ser esencial para que el decodificador decodifique las imágenes de video del flujo de bits, pero el decodificador puede usar la información para mejorar la visualización o el procesamiento de las imágenes (por ejemplo, la salida decodificada). La información de un mensaje SEI puede ser metadatos incrustados. En un ejemplo ilustrativo, las entidades del lado del decodificador podrían utilizar la información de un mensaje SEI para mejorar la visibilidad del contenido. En algunos casos, ciertos estándares de aplicación pueden exigir la presencia de dichos mensajes SEI en el flujo de bits para que la mejora de la calidad se pueda llevar a todos los dispositivos que cumplen con el estándar de la aplicación (por ejemplo, el transporte del mensaje SEI de empaquetamiento de cuadros para el formato de vídeo 3DTV plano-estereoscópico compatible con los cuadros, en donde el mensaje SEI se transporta para cada cuadro del video, el manejo de un mensaje SEI de punto de recuperación, el uso del mensaje SEI de rectángulo de barrido panorámico en DVB, además de muchos otros

ejemplos).

[0067] En algunas formas de realización, el dispositivo de codificación de video 104 y/o el dispositivo de decodificación de video 112 pueden integrarse con un dispositivo de codificación de audio y un dispositivo de decodificación de audio, respectivamente. El dispositivo de codificación de video 104 y/o el dispositivo de decodificación de video 112 también pueden incluir otro hardware o software que sea necesario para implementar las técnicas de codificación descritas con anterioridad, como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables en campo (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. El dispositivo de codificación de video 104 y el dispositivo de decodificación de video 112 pueden integrarse como parte de un codificador/decodificador combinado (códec) en un dispositivo respectivo. Un ejemplo de detalles específicos del dispositivo de codificación 104 se describe a continuación con referencia a la Figura 7. Un ejemplo de detalles específicos del dispositivo de decodificación 112 se describe a continuación con referencia a la Figura 8.

[0068] El sistema de ejemplo mostrado en la Figura 1 es un ejemplo ilustrativo que se puede utilizar aquí. Las técnicas para procesar datos de video usando las técnicas descritas en este documento pueden realizarse mediante cualquier dispositivo de codificación y/o decodificación de video digital. Aunque generalmente las técnicas de la presente divulgación se realizan mediante un dispositivo de codificación de video o un dispositivo de decodificación de video, las técnicas también se pueden realizar mediante un codificador-decodificador de video combinado, normalmente denominado "CODEC". Además, las técnicas de la presente divulgación también pueden ser realizadas por un preprocesador de video. El dispositivo de origen y el dispositivo receptor son simplemente ejemplos de tales dispositivos de codificación en los que el dispositivo de origen genera datos de video codificados para su transmisión al dispositivo receptor. En algunos ejemplos, los dispositivos de origen y receptor pueden operar de una manera sustancialmente simétrica de manera que cada uno de los dispositivos incluya componentes de codificación y decodificación de video. Por lo tanto, los sistemas de ejemplo pueden soportar la transmisión de video unidireccional o bidireccional entre dispositivos de video, por ejemplo, para transmisión de video, reproducción de video, transmisión de video o telefonía de video.

[0069] Como se describió con anterioridad, para cada bloque, puede estar disponible un conjunto de información de movimiento (también denominado aquí como parámetros

de movimiento). Un conjunto de información de movimiento contiene información de movimiento para direcciones de predicción hacia adelante y hacia atrás. Las direcciones de predicción hacia adelante y hacia atrás son dos direcciones de predicción de un modo de predicción bidireccional, en cuyo caso los términos “hacia adelante” y “hacia atrás” no tienen necesariamente un significado geométrico. En cambio, “adelante” y “atrás” corresponden a la lista de imágenes de referencia 0 (RefPicList0 o L0) y la lista de imágenes de referencia 1 (RefPicList1 o L1) de una imagen actual. En algunos ejemplos, cuando solo hay disponible una lista de imágenes de referencia para una imagen o un segmento, solo RefPicList0 está disponible y la información de movimiento de cada bloque de un segmento siempre está hacia adelante.

[0070] En algunos casos, un vector de movimiento junto con su índice de referencia se usa en procesos de codificación (por ejemplo, compensación de movimiento). Un vector de movimiento de este tipo con el índice de referencia asociado se indica como un conjunto unipredictivo de información de movimiento. Para cada dirección de predicción, la información de movimiento puede contener un índice de referencia y un vector de movimiento. En algunos casos, por simplicidad, un vector de movimiento en sí mismo puede ser referido de una manera que se asume que tiene un índice de referencia asociado. Se utiliza un índice de referencia para identificar una imagen de referencia en la lista de imágenes de referencia actual (RefPicList0 o RefPicList1). Un vector de movimiento tiene un componente horizontal y uno vertical que proporcionan un desplazamiento desde la posición de las coordenadas en la imagen actual hasta las coordenadas en la imagen de referencia identificadas por el índice de referencia. Por ejemplo, un índice de referencia puede indicar una imagen de referencia particular que debe usarse para un bloque en una imagen actual, y el vector de movimiento puede indicar en qué parte, en la imagen de referencia, el bloque que mejor coincide (el bloque que mejor coincide con el bloque actual) está en la imagen de referencia.

[0071] Se puede utilizar un recuento de orden de imágenes (POC) en estándares de codificación de video para identificar un orden de visualización de una imagen. Aunque hay casos en los que dos imágenes dentro de una secuencia de video codificada pueden tener el mismo valor de POC, normalmente no ocurre dentro de una secuencia de video codificada. Cuando están presentes múltiples secuencias de video codificadas en un flujo de bits, las imágenes con un mismo valor de POC pueden estar más cerca entre sí en términos de orden de decodificación. Los valores POC de las imágenes se pueden utilizar para la construcción de la lista de imágenes de referencia, la derivación del conjunto de imágenes de referencia como en HEVC y la escala de vectores de

movimiento.

[0072] En H.264/AVC, cada intermacrobloque (MB) puede dividirse de cuatro formas diferentes, que incluyen: una partición de 16x16 MB; dos particiones de 16x8 MB; dos particiones de 8x16 MB; y cuatro particiones de 8x8 MB. Diferentes particiones MB en un MB pueden tener diferentes valores de índice de referencia para cada dirección (RefPicList0 o RefPicList1). En algunos casos, cuando un MB no está dividido en cuatro particiones de 8x8 MB, solo puede tener un vector de movimiento para cada partición MB en cada dirección. En algunos casos, cuando un MB se divide en cuatro particiones de 8x8 MB, cada partición de 8x8 MB se puede dividir en subbloques, en cuyo caso cada subbloque puede tener un vector de movimiento diferente en cada dirección. En algunos ejemplos, hay cuatro formas diferentes de obtener subbloques de una partición de 8x8 MB, que incluyen: un subbloque de 8x8; dos subbloques de 8x4; dos subbloques de 4x8; y cuatro subbloques 4x4. Cada subbloque puede tener un vector de movimiento diferente en cada dirección. Por lo tanto, un vector de movimiento está presente en un nivel igual a superior al subbloque.

[0073] En AVC, se puede habilitar un modo directo temporal en el nivel de MB o en el nivel de partición de MB para el modo de salto y/o directo en segmentos B. Para cada partición MB, los vectores de movimiento del bloque coubicado con la partición MB actual en RefPicList1 [0] del bloque actual se utilizan para derivar los vectores de movimiento. Cada vector de movimiento en el bloque coubicado se escala en función de las distancias POC.

[0074] También se puede realizar un modo directo espacial en AVC. Por ejemplo, en AVC, un modo directo también puede predecir la información de movimiento de los vecinos espaciales.

[0075] En HEVC, la unidad de codificación más grande en un segmento se denomina bloque de árbol de codificación (CTB). Un CTB contiene un árbol cuádruple, cuyos nodos son unidades de codificación. El tamaño de un CTB puede variar de 16x16 a 64x64 en el perfil principal HEVC. En algunos casos, se pueden admitir tamaños CTB de 8x8. Una unidad de codificación (CU) podría ser del mismo tamaño que un CTB y tan pequeña como 8x8. En algunos casos, cada unidad de codificación se codifica con un modo. Cuando una CU está intercodificada, la CU puede dividirse en 2 o 4 unidades de predicción (PU), o puede convertirse en una sola PU cuando no se aplica más partición. Cuando dos PU están presentes en una CU, pueden ser rectángulos de la mitad del tamaño o dos rectángulos con un tamaño de $\frac{1}{4}$ o $\frac{3}{4}$ de la CU.

[0076] Cuando la CU está intercodificada, un conjunto de información de movimiento

está presente para cada PU. Además, cada PU está codificada con un modo de interpredicción único para derivar el conjunto de información de movimiento.

[0077] Para la predicción de movimiento en HEVC, hay dos modos de interpredicción, incluido el modo de fusión y el modo de predicción de vector de movimiento avanzado (AMVP) para una unidad de predicción (PU). Omitir se considera un caso especial de fusión. En el modo AMVP o de fusión, se mantiene una lista de candidatos de vector de movimiento (MV) para múltiples predictores de vector de movimiento. El vector o los vectores de movimiento, así como los índices de referencia en el modo de fusión, de la PU actual se generan tomando un candidato de la lista de candidatos MV. En algunos ejemplos, como se describe a continuación, se pueden incluir uno o más indicadores de compensación de iluminación local (LIC) almacenados junto con vectores de movimiento almacenados en una lista de candidatos de MV.

[0078] En los ejemplos en los que se usa una lista de candidatos de MV para la predicción de movimiento (y cuando sea aplicable, la compensación de iluminación) de un bloque, el dispositivo de codificación y el dispositivo de decodificación pueden construir la lista de candidatos de MV por separado. Por ejemplo, la lista de candidatos MV puede ser generada por un dispositivo de codificación cuando se codifica un bloque, y puede ser generada por un dispositivo de decodificación cuando se decodifica el bloque. La información relacionada con los candidatos de información de movimiento en la lista de candidatos de MV (por ejemplo, información relacionada con uno o más vectores de movimiento, información relacionada con uno o más indicadores de LIC que se pueden almacenar en la lista de candidatos de MV en algunos casos y/u otra información), se puede señalar entre el dispositivo de codificación y el dispositivo de decodificación. Por ejemplo, en el modo de fusión, los valores de índice de los candidatos de información de movimiento almacenados se pueden señalar desde un dispositivo de codificación a un dispositivo de decodificación (por ejemplo, en una estructura de sintaxis, como el conjunto de parámetros de imagen (PPS), el conjunto de parámetros de secuencia (SPS)), conjunto de parámetros de video (VPS), un encabezado de segmento, un mensaje de información de mejora suplementaria (SEI) enviado dentro o por separado del flujo de bits de video y/u otra señalización). El dispositivo de decodificación puede construir una lista de candidatos de MV y usar las referencias o índices señalados para obtener uno o más candidatos de información de movimiento de la lista de candidatos de MV construida para usar para la predicción de compensación de movimiento. Por ejemplo, el dispositivo de decodificación puede construir una lista de candidatos MV y usar un vector de movimiento (y en algunos casos

un indicador LIC) desde una ubicación indexada para la predicción de movimiento del bloque. En el caso del modo AMVP, además de las referencias o índices, las diferencias o valores residuales también se pueden señalar como deltas. Por ejemplo, para el modo AMVP, el dispositivo de decodificación puede construir una o más listas de candidatos MV y aplicar los valores delta a uno o más candidatos de información de movimiento obtenidos usando los valores de índice señalados al realizar la predicción de compensación de movimiento del bloque.

[0079] En algunos ejemplos, la lista de candidatos MV contiene hasta cinco candidatos para el modo de fusión y dos candidatos para el modo AMVP. En otros ejemplos, se pueden incluir diferentes números de candidatos en una lista de candidatos MV para el modo de fusión y/o el modo AMVP. Un candidato de fusión puede contener un conjunto de información de movimiento. Por ejemplo, un conjunto de información de movimiento puede incluir vectores de movimiento correspondientes tanto a las listas de imágenes de referencia (lista 0 y lista 1) como a los índices de referencia. Si un candidato de fusión se identifica mediante un índice de fusión, las imágenes de referencia se utilizan para la predicción de los bloques actuales, así como también se determinan los vectores de movimiento asociados. Sin embargo, en el modo AMVP, para cada dirección de predicción potencial de la lista 0 o la lista 1, se debe señalar explícitamente un índice de referencia, junto con un índice MVP a la lista de candidatos MV, ya que el candidato AMVP contiene solo un vector de movimiento. En el modo AMVP, los vectores de movimiento predichos se pueden refinar aún más.

[0080] Como puede verse con anterioridad, un candidato de fusión corresponde a un conjunto completo de información de movimiento, mientras que un candidato de AMVP contiene solo un vector de movimiento para una dirección de predicción y un índice de referencia específicos. Los candidatos para ambos modos se derivan de manera similar de los mismos bloques vecinos espaciales y temporales.

[0081] En algunos ejemplos, el modo de fusión permite que una PU interpredicha herede el mismo vector o vectores de movimiento, dirección de predicción e índice o índices de imagen de referencia de una PU interpredicha que incluye una posición de datos de movimiento seleccionada de un grupo de posiciones de datos de movimiento espacialmente vecinas y una de las dos posiciones de datos de movimiento coubicadas temporalmente. Para el modo AMVP, el vector de movimiento o los vectores de una PU pueden codificarse de forma predicativa con respecto a uno o más predictores de vector de movimiento (MVP) de una lista de candidatos AMVP construida por un codificador y/o un decodificador. En algunos casos, para la interpredicción en una sola dirección de

- una PU, el codificador y/o decodificador pueden generar una única lista de candidatos AMVP. En algunos casos, para la predicción bidireccional de una PU, el codificador y/o decodificador pueden generar dos listas de candidatos AMVP, una usando datos de movimiento de PU vecinas espaciales y temporales desde la dirección de predicción directa y otra usando datos de movimiento de PU vecinas espaciales y temporales desde la dirección de predicción hacia atrás.
- [0082]** Los candidatos para ambos modos pueden derivarse de bloques vecinos espaciales y/o temporales. Por ejemplo, la Figura 2A y Figura 2B incluyen diagramas conceptuales que ilustran candidatos vecinos espaciales en HEVC.
- [0083]** La Figura 2A ilustra candidatos de vector de movimiento vecino espacial (MV) para el modo de fusión. La Figura 2B ilustra candidatos de vector de movimiento vecino espacial (MV) para el modo AMVP. Los candidatos de MV espacial se derivan de los bloques vecinos para una PU específica (PU0), aunque los métodos que generan los candidatos a partir de los bloques difieren para los modos de fusión y AMVP.
- [0084]** En el modo de fusión, el codificador y/o decodificador pueden formar una lista de candidatos de fusión considerando candidatos de fusión de varias posiciones de datos de movimiento. Por ejemplo, como se muestra en la Figura 2A, se pueden derivar hasta cuatro candidatos MV espaciales con respecto a las posiciones de datos de movimiento espacialmente vecinas mostradas con los números 0-4 en la Figura 2A. Los candidatos a MV se pueden ordenar en la lista de candidatos combinados en el orden indicado por los números 0-4. Por ejemplo, las posiciones y el orden pueden incluir: posición izquierda (0), posición superior (1), posición superior derecha (2), posición inferior izquierda (3) y posición superior izquierda (4).
- [0085]** En el modo AMVP mostrado en la Figura 2B, los bloques vecinos se dividen en dos grupos: el grupo de la izquierda incluye los bloques 0 y 1, y el grupo superior incluye los bloques 2, 3 y 4. Para cada grupo, el candidato potencial en un bloque vecino se refiere a la misma imagen de referencia ya que el indicado por el índice de referencia señalado tiene la máxima prioridad para ser elegido para formar un candidato final del grupo. Es posible que todos los bloques vecinos no contengan un vector de movimiento que apunte a la misma imagen de referencia. Por lo tanto, si no se puede encontrar dicho candidato, el primer candidato disponible se escalará para formar el candidato final, por lo que se pueden compensar las diferencias de distancia temporal.
- [0086]** Algún ejemplo descrito en el presente documento puede usar un modo de resolución de vector de movimiento adaptativo (AMVR), como se indicó con anterioridad.
- AMVR se refiere al ajuste adaptativo de un rango o resolución de un vector de

movimiento durante el procesamiento de imágenes. El uso de AMVR permite diferentes niveles de precisión en la codificación de las diferencias del vector de movimiento, lo que permite mejorar la eficiencia de codificación y puede implicar el uso de unidades de muestra fraccionarias (por ejemplo, medio pel u otras muestras fraccionarias similares en cualquier nivel de resolución). En algunas implementaciones, se puede usar una unidad de muestra fraccionaria uniforme, una unidad de muestra entera o similar en todo el rango, o el rango de la unidad de muestra se puede establecer de forma adaptativa según la región. El primero puede denominarse AMVR lineal y el último puede denominarse AMVR no lineal. El uso de un modo AMVR puede implicar además la señalización explícita de un valor de índice de filtro (por ejemplo, usando un elemento de sintaxis en un flujo de bits de video) para su uso en el procesamiento de compensación de movimiento de una imagen actual.

[0087] En el contexto descrito con anterioridad, el remuestreo de imagen de referencia (RPR) es un proceso de remuestreo que puede usarse para permitir imágenes de referencia con un tamaño de imagen que es diferente del tamaño de imagen de una imagen actual que se está codificando o decodificando. Cuando una imagen de referencia tiene un tamaño diferente al de la imagen actual, se puede invocar un proceso de remuestreo de imagen para proporcionar versiones con sobremuestreo o inferior de una de las imágenes para que coincida con el tamaño de la otra imagen (por ejemplo, sobremuestreo o submuestreo de la imagen de referencia para que coincida el tamaño actual de la imagen). Este remuestreo puede, en algunos ejemplos, ser similar a las operaciones de escalabilidad espacial utilizadas en la extensión escalable (SHVC) del estándar H.265/HEVC. El proceso RPR se puede aplicar a imágenes de una sola capa que tienen diferentes tamaños y/o se puede aplicar a imágenes de diferentes capas que tienen diferentes tamaños. Un ejemplo ilustrativo de la aplicación de RPR a imágenes de diferentes tamaños de diferentes capas se describe a continuación con respecto a la Figura 3.

[0088] La Figura 3 es un diagrama que ilustra aspectos de los datos de imagen en un sistema a diferentes valores de escala (por ejemplo, diferentes tamaños). Las imágenes o los datos de las imágenes (por ejemplo, unidades de acceso) se pueden estructurar en capas. Cada capa puede ser para cuadros de imágenes con diferentes tamaños (por ejemplo, resoluciones). Un ejemplo de este tipo ilustrado en la Figura 3 incluye dos capas, y otros varios ejemplos pueden usar otros números de capas (por ejemplo, tres capas, cuatro capas, etc.). Como se señaló con anterioridad, las imágenes dentro de una sola capa pueden tener diferentes tamaños, en cuyo caso se puede realizar el

proceso de RPR. En el ejemplo de la Figura 3, se muestran los datos de tres imágenes, con cada imagen asociada con dos tamaños de imagen diferentes. Una primera imagen está asociada con la unidad 314 de capa 1 en un primer tamaño de imagen, y la unidad 312 de capa 0 en un segundo tamaño de imagen. De manera similar, una segunda imagen está asociada con la unidad 324 de capa 1 en el primer tamaño de imagen, y la unidad 322 de capa 0 en el segundo tamaño de imagen, y una tercera imagen está asociada con una unidad 334 de capa 1 en el primer tamaño de imagen y unidad de capa 0 332 en el segundo tamaño de imagen. Durante la señalización, se pueden construir listas de imágenes de referencia para identificar imágenes de referencia para su uso en el procesamiento de una imagen actual. En algunos ejemplos, las imágenes de una capa (por ejemplo, en la primera resolución) se pueden utilizar como imágenes de referencia para una imagen actual en una segunda capa (por ejemplo, en la segunda resolución). En otro ejemplo, una sola capa puede tener imágenes de diferentes tamaños, como se indicó con anterioridad.

15 **[0089]** Con RPR, las imágenes (o bloques de las imágenes) de tamaño diferente (por ejemplo, imágenes de una sola capa o imágenes de diferentes capas) se pueden usar como imágenes de referencia para bloques de la imagen actual. Tales imágenes de referencia se pueden almacenar en una memoria intermedia de imágenes decodificadas (DPB) (por ejemplo, usando memoria o almacenamiento de imágenes de referencia como la memoria de imágenes 92 de la Figura 8 para una unidad de filtro tal como la unidad de filtro 91 de la Figura 8). Se puede utilizar una herramienta de muestreo de imágenes de referencia para generar los datos de referencia necesarios a partir de la imagen de referencia para procesar la imagen actual. En algunos ejemplos, esta herramienta de muestreo de imágenes de referencia puede ser parte de una unidad de filtro (por ejemplo, la unidad de filtro 91 de la Figura 8), o en otros ejemplos puede ser parte de cualquier aspecto de un dispositivo para codificar o decodificar como se describe en este documento.

20 **[0090]** La Figura 4 es un diagrama que ilustra aspectos adicionales del remuestreo de imágenes de referencia de acuerdo con los ejemplos descritos en el presente documento. La Figura 4 incluye una imagen de referencia 410 y una imagen actual 420. La imagen de referencia 410 tiene un primer espaciado de cuadrícula horizontal 414 y un primer espaciado de cuadrícula vertical 412. La imagen actual 420 tiene un segundo espaciado de cuadrícula horizontal 424 que es diferente del primer espaciado de cuadrícula horizontal 414. La imagen actual también tiene un segundo espaciado de cuadrícula vertical 422 que es diferente del primer espaciado de cuadrícula vertical 412.

30

35

Los primeros espaciados de cuadrícula 412 y 414 pueden, por ejemplo, asociarse con una primera capa, y los segundos espaciados de cuadrícula 422 y 424 pueden estar asociado con una segunda capa. La imagen de referencia 410 puede ser una imagen de la primera capa que es secuencial con la imagen actual 420 de la segunda capa en un flujo de video, con la imagen de referencia 410 seleccionada como imagen de referencia para la imagen actual 420 como parte de las operaciones de predicción.

[0091] La imagen de referencia 410 y la imagen actual 420 incluyen cuadrículas de píxeles, incluidos los píxeles 418 y 419 de la imagen de referencia 410 y el píxel 428 de la imagen actual 420. Cada píxel puede tener datos asociados, tales como valores de color para porciones rojas, verdes y azules del píxel. Cada color tendrá un valor de datos en cada píxel. Cada píxel de cada imagen se puede asociar con una posición entera en la cuadrícula (por ejemplo, el tamaño o el valor de escala) de la imagen. Un vector de movimiento estándar 430 que apunta directamente de un píxel a otro puede usar las posiciones enteras de los píxeles. Sin embargo, la precisión fraccionaria, como se describió con anterioridad, puede permitir que el vector de movimiento 430 apunte entre píxeles. La precisión de medio pel podría, por ejemplo, permitir que el vector de movimiento 430 apunte a la mitad entre el píxel 419 y 418. La precisión de un cuarto de pel permite que el vector de movimiento 430 apunte a tres posiciones diferentes entre los píxeles (por ejemplo, un cuarto, la mitad y tres cuartos), así como directamente a las posiciones enteras de los píxeles.

[0092] El AMVR, como se describió con anterioridad, permite que la precesión del vector de movimiento se establezca de forma adaptativa en diferentes niveles de precisión fraccionaria. Cuando se implementa el procesamiento de imágenes usando un modo AMVR, la precisión de MV se puede seleccionar de forma adaptativa con una precisión de pel enteros (por ejemplo, en donde el MV está en los puntos de la cuadrícula solo para los espaciados de cuadrícula 412, 414, 422 y 424, sin precisión fraccionaria), o en cualquier nivel de precisión fraccionaria respaldado por un modo AMVR implementado (por ejemplo, permitir que el MV apunte entre los puntos de la cuadrícula con un nivel de precisión fraccionaria para los espaciados de cuadrícula 412, 414, 422 y 424).

Permitir que un sistema determine de forma adaptativa un nivel de resolución (por ejemplo, resolución de números enteros en el nivel de espaciado de la cuadrícula o una resolución fraccionada para la precisión entre los espaciados de la cuadrícula) permite mejorar la eficiencia de codificación. Esta eficiencia de codificación mejorada puede usar elementos de sintaxis adicionales para la señalización, que pueden indicar además explícitamente un índice de filtro para una imagen actual, como se describe para varios

ejemplos en este documento.

[0093] La interpolación se usa para seleccionar los valores de datos (por ejemplo, para rojo, verde y azul) en la posición del píxel dado el movimiento esperado u otros cambios, incluidos los datos intrapíxel (por ejemplo, de un vector de movimiento usando precisión fraccionaria). Durante el procesamiento (por ejemplo, decodificación o codificación), el procesamiento se realiza para cada píxel o para cada muestra (por ejemplo, una muestra de luminancia y/o muestra de crominancia) de cada píxel. El filtrado se realiza para cada píxel o cada muestra de cada píxel, incluida la interpolación y cualquier compensación de remuestreo debida a RPR, para generar píxeles o valores de muestra para la imagen actual 420 durante el procesamiento de datos de video. En otro ejemplo, el filtrado de interpolación y remuestreo se pueden fusionar en un proceso de filtrado. Cada píxel o muestra tendrá un valor de escala adjunto y un vector de movimiento durante el procesamiento. Como se describió con anterioridad, se pueden usar diferentes filtros en distintas circunstancias (por ejemplo, diferentes valores del vector de movimiento 430 o diferentes valores de escala entre la imagen de referencia 410 y la imagen actual 420).

[0094] La tabla 1 a continuación proporciona una tabla de ejemplo de coeficientes de filtro con un índice de filtro para la conmutación del filtro de interpolación entre diferentes filtros de interpolación de medio pel:

Posición de muestra fraccionada	coeficientes de filtro de interpolación							
	A	B	C	D	E	F	G	H
1	0	1	-3	63	4	-2	1	0
2	-1	2	-5	62	8	-3	1	0
3	-1	3	-8	60	13	-4	1	0
4	-1	4	-10	58	17	-5	1	0
5	-1	4	-11	52	26	-8	3	-1
6	-1	3	-9	47	31	-10	4	-1
7	-1	4	-11	45	34	-10	4	-1
8 (Valor de índice de filtro 1)	-1	4	-11	40	40	-11	4	-1
8 (Valor de índice de filtro 2)	0	3	9	20	20	9	3	0
9	-1	4	-10	34	45	-11	4	-1
10	-1	4	-10	31	47	-9	3	-1
11	-1	3	-8	26	52	-11	4	-1
12	0	1	-5	17	58	-10	4	-1
13	0	1	-4	13	60	-8	3	-1
14	0	1	-3	8	62	-5	2	-1
15	0	1	-2	4	63	-3	1	0

Tabla 1

[0095] La Tabla 1 muestra una tabla de coeficientes de filtro de interpolación para filtrar

en posiciones de precisión fraccionaria de un dieciseisavo. Dependiendo de la posición de precisión fraccionaria (por ejemplo, posición de muestra fraccionaria 1-15 en la primera columna, siendo la posición 0 la posición no fraccionaria o entera), se pueden usar diferentes coeficientes de filtro (por ejemplo, coeficientes de la fila correspondiente a un posición fraccionaria). Además, en la posición de medio pel (por ejemplo, la posición de muestra fraccionaria 8 en la precisión de una décimo sexta fracción de la Tabla 1), hay dos filtros de interpolación disponibles y la señalización se puede usar para cambiar entre los filtros en este nivel. Si bien la Tabla 1 describe dos filtros para la conmutación del filtro de interpolación en el nivel de medio pel (por ejemplo, seleccionando entre los valores de índice de filtro 1 y 2), otros ejemplos pueden usar más de dos filtros para la conmutación o pueden incluir filtros adicionales para la conmutación en cualquier posición fraccionaria. La adición de filtros adicionales para la conmutación del filtro de interpolación puede aumentar la precisión fraccionaria durante el procesamiento de imágenes a costa de recursos de procesamiento y señalización adicionales.

[0096] En JVET-N0279 y JVET-O0242 se propone RPR. En la propuesta, una imagen actual (por ejemplo, la imagen que se está codificando o decodificando actualmente) y las imágenes de referencia pueden tener diferentes resoluciones (por ejemplo, tamaños, espaciados de cuadrícula, valores de escala, etc.) como se describió con anterioridad. En tales casos, durante la compensación de movimiento, se considera una diferencia en la resolución al realizar el filtrado de interpolación ajustando la fase del filtro de interpolación. Por ejemplo, en la comunicación de video en tiempo real, permitir cambios de resolución dentro de una secuencia de video codificada sin insertar una imagen intrapredicha (imagen I) no solo puede adaptar los datos de video a las condiciones dinámicas del canal o las preferencias del usuario sin problemas, sino que también puede eliminar los efectos causados por imágenes I. Cuando se producen cambios de resolución adaptativa, se puede predecir una imagen actual a partir de imágenes de referencia de diferentes tamaños. A continuación, se describe un ejemplo ilustrativo de un proceso de remuestreo (o escalado) de imagen de referencia.

[0097] En JVET-O0057 se propuso a VVC un concepto de filtro de interpolación conmutable. En este documento, se pueden cambiar uno o más filtros de interpolación. Por ejemplo, un índice de filtro puede ser señalado o inferido (por ejemplo, por el decodificador) para indicar que se usa otro filtro de interpolación para un bloque. El índice de filtro se puede señalar en bloque (por ejemplo, explícitamente o mediante señalización en modo AMVR de medio pel). Por ejemplo, en un sistema con AMVR de

medio pel, los elementos de sintaxis señalados en un flujo de bits pueden indicar explícitamente un índice de filtro que se utilizará en el procesamiento de un bloque actual asociado con el elemento de sintaxis (por ejemplo, un elemento de sintaxis que indica un valor de índice de filtro particular de la Tabla 1). En ejemplos en los que AMVR

5 utiliza una precisión fraccionaria distinta a la de medio pel, una señalización similar puede utilizar elementos de sintaxis configurados para las opciones de resolución particulares de la implementación de AMVR.

[0098] En otras implementaciones, el índice de filtro no necesita ser señalado explícitamente a través de elementos de sintaxis en el flujo de bits que indiquen específicamente el índice de filtro que se utilizará. En tales otros ejemplos sin

10 señalización explícita, el índice de filtro se puede inferir (por ejemplo, mediante el decodificador) a través de un vector de movimiento candidato en los modos de fusión, como los modos de fusión descritos con anterioridad con respecto a las Figuras 2A y 2B.

15 **[0099]** En un ejemplo ilustrativo, se puede definir un total de N filtros con diferentes características (por ejemplo, como la longitud del filtro, las frecuencias de corte, la banda de transición o la cantidad de ondulaciones). Para cada bloque, se puede definir un subconjunto de los N filtros y el índice del filtro elegido en el subconjunto se puede señalar explícita o implícitamente al decodificador (por ejemplo, usando un indicador o

20 un valor de índice de filtro como se ilustra en la Tabla 1). El subconjunto puede variar de un bloque a otro en función de cierto criterio (por ejemplo, relación de aciertos de filtros en bloques previamente codificados, valores de vector de movimiento, valores de escala, etc.).

[0100] Cuando se produce la conmutación del filtro de interpolación, el filtro indicado por el índice de filtro puede ser adecuado para propósitos de interpolación, pero puede

25 no ser útil para propósitos de remuestreo de imagen (y en algunos casos otros procesos de conversión de frecuencia de muestreo). Por ejemplo, un filtro de suavizado puede no ser un buen filtro de remuestreo. El cambio entre diferentes filtros (por ejemplo, cambio de filtro de interpolación) permite que las operaciones de procesamiento de una imagen creen estabilidad entre las imágenes dependiendo de las características de un píxel que

30 se está filtrando (por ejemplo, vectores de movimiento asociados y valores de escala para un píxel en una imagen actual y una imagen de referencia asociada). Por ejemplo, un índice de filtro puede tener valores de índice de filtro para diferentes filtros. En algunos ejemplos, el filtro que se utilizará se señala (por ejemplo, mediante un

35 indicador) para indicar cuál de los filtros del índice se utilizará (por ejemplo, cambiar

entre filtros de interpolación). En otro ejemplo, el filtro que se usará puede provenir de un candidato espacial (por ejemplo, una imagen de referencia) donde el índice de filtro se almacena con datos de vectores de movimiento y se propaga desde una imagen de referencia para usar con una imagen actual.

- 5 **[0101]** En el presente documento se describen métodos, aparatos y medios legibles por computadora que abordan el problema identificado con anterioridad, así como otros problemas. En algunos ejemplos, como se describe con más detalle a continuación, se puede realizar una verificación de condición para identificar si un tamaño de imagen actual y un tamaño de imagen de referencia son del mismo tamaño. La verificación del
10 estado se puede realizar en diferentes niveles, como se detalla a continuación.

[0102] La Figura 5 es un diagrama de flujo que ilustra un método 500 para procesar datos de video. En algunos ejemplos, el método 500 se realiza mediante un dispositivo que incluye una memoria y uno o más procesadores. En algunos ejemplos, el método 500 puede realizarse mediante un dispositivo de codificación, como el dispositivo 104.

- 15 En algunos ejemplos, el método 500 puede realizarse mediante un dispositivo de decodificación, como el dispositivo 112. En algunos ejemplos, el método 500 se materializa como instrucciones almacenadas en almacenamiento legible por computadora que hace que uno o más procesadores realicen el método 500 cuando las instrucciones son ejecutadas por uno o más procesadores.

- 20 **[0103]** En la operación 505, el método 500 incluye obtener una imagen actual y una imagen de referencia. Como se describió con anterioridad, en varios ejemplos, la imagen actual y la imagen de referencia se pueden dividir y organizar de varias formas (por ejemplo, varias unidades, segmentos, etc.).

- [0104]** La operación 510 del método 500 incluye la identificación de información de
25 índice de filtro para un bloque actual de la imagen actual. En un ejemplo ilustrativo, esta información de índice de filtro se puede inferir de bloques vecinos. De acuerdo a las reivindicaciones adjuntas, se señala como parte de AMVR. Específicamente, la información del índice de filtro comprende datos de señalización del modo AMVR donde el índice del filtro actual se señala a través de la señalización del modo AMVR. En
30 algunos ejemplos ilustrativos, la información de índice de filtro comprende información de bloque vecino y el índice de filtro actual se deriva entonces de la información de bloque vecino durante una derivación de lista de candidatos de vectores de movimiento en modo de fusión. La información del índice de filtro incluye información de movimiento, como un índice de filtro o un valor de índice de filtro. En un modo de fusión, esta
35 información de índice de filtro se puede propagar o implicar desde bloques vecinos. Con

un modo AMVR, la información del índice de filtro se puede señalar explícitamente mediante la señalización AMVR.

[0105] La operación 515 del método 500 incluye determinar que un primer valor de tamaño de imagen de la imagen actual y un segundo valor de tamaño de imagen de la imagen de referencia son diferentes. En algunos ejemplos, esta determinación puede implicar una comparación de ancho y alto. En otros ejemplos, se puede utilizar cualquier otra comparación de este tipo. En un ejemplo, determinar que el primer valor de tamaño de imagen de la imagen actual y el segundo valor de tamaño de imagen de la imagen de referencia son diferentes comprende determinar al menos uno de: la anchura de la imagen actual es diferente de la anchura de la imagen de referencia; y la altura de la imagen actual es diferente de la altura de la imagen de referencia. Esta verificación del tamaño de la imagen puede funcionar como una condición para cambiar el índice de filtro. Dependiendo de la verificación del tamaño de la imagen y un cambio asociado del índice de filtro, la propagación del valor del índice original (por ejemplo, el valor del índice actual antes de ser reemplazado si lo indica la verificación del tamaño de la imagen) se puede configurar (por ejemplo, almacenando el valor de índice original y reemplazarlo por un paso de propagación, o manteniéndolo si un valor predeterminado no reemplaza el valor de índice original).

[0106] En la operación 520, el método 500 incluye realizar un proceso de conversión de frecuencia de muestreo usando un índice de filtro predeterminado en lugar de un índice de filtro actual identificado por la información del índice de filtro. En algunos ejemplos, el proceso de conversión de la frecuencia de muestreo incluye un proceso de remuestreo. Por ejemplo, la operación 520 puede incluir realizar un proceso de remuestreo usando un índice de filtro predeterminado en lugar de un índice de filtro actual identificado por la información del índice de filtro. En algunos ejemplos, el proceso de conversión de la frecuencia de muestreo incluye un proceso de interpolación. La interpretación se puede basar en la determinación de que el primer valor de tamaño de imagen de la imagen actual y el segundo valor de tamaño de imagen de la imagen de referencia son diferentes. En varios ejemplos, el proceso de conversión de la frecuencia de muestreo (por ejemplo, el proceso de remuestreo) se realiza sin usar el índice de filtro actual cuando las imágenes actual y de referencia tienen valores de tamaño diferentes. En algunos de estos ejemplos, el valor predeterminado para el índice de filtro actual es interpretado por un dispositivo o incluye una indicación de que el filtrado conmutable alternativo no se aplica como parte del proceso de conversión de la frecuencia de muestreo (por ejemplo, el proceso de remuestreo).

[0107] En varios ejemplos, cuando el índice de filtro derivado (por ejemplo, el índice actual indicado por señalización o derivación), el valor permanece con el bloque y puede usarse (por ejemplo, propagarse) a bloques posteriores procesados por el sistema. En tales ejemplos, si se usa un valor de índice de filtro predeterminado para una imagen actual (por ejemplo, no se usa el filtro conmutable alternativo), el valor de índice de filtro predeterminado no se propaga desde la imagen actual, pero el valor original se puede almacenar o el valor predeterminado se puede reemplazar con el valor original después de su uso para la propagación a las imágenes posteriores.

[0108] En algunos ejemplos, el método puede involucrar además operaciones para almacenar el índice de filtro actual identificado por la información del índice de filtro con el bloque actual y usar el índice de filtro actual en la derivación del candidato de vector de movimiento para un bloque posterior basado en el índice de filtro actual, siendo identificado por la información del índice de filtro. En algunos ejemplos, el método puede involucrar operaciones para asignar un valor predeterminado para el índice de filtro actual para reemplazar un valor original de manera temporal cuando el proceso de conversión de frecuencia de muestreo (por ejemplo, el proceso de remuestreo) se realiza sin usar el índice de filtro actual para crear el índice de filtro predeterminado y asignar el valor original para el índice de filtro actual después del proceso de conversión de frecuencia de muestreo (por ejemplo, el proceso de remuestreo). Esta reasignación del original permite que el valor del índice del filtro actual (por ejemplo, el valor original) se propague a las imágenes posteriores, incluso cuando el valor del índice del filtro actual no se utiliza para la imagen actual.

[0109] Algunos de estos ejemplos pueden funcionar determinando que el índice de filtro actual no se utiliza para el proceso de conversión de frecuencia de muestreo (por ejemplo, el proceso de remuestreo) y reemplazando un valor del índice de filtro actual con un valor predeterminado en respuesta a la determinación de que el índice de filtro actual no se utiliza para el proceso de conversión de la frecuencia de muestreo (por ejemplo, el proceso de remuestreo). De manera similar, algunos de estos ejemplos funcionan procesando el valor predeterminado para determinar que no se aplica un filtro conmutable alternativo.

[0110] Algunos ejemplos pueden funcionar donde un valor de índice de filtro actual no se señala en respuesta a una determinación de que el índice de filtro actual no se usa en el proceso de conversión de frecuencia de muestreo (por ejemplo, el proceso de remuestreo). Algunos ejemplos pueden funcionar cuando el índice de filtro actual se selecciona de un conjunto de filtros, en cuyo caso se utiliza un subconjunto del conjunto

de filtros para realizar el proceso de conversión de frecuencia de muestreo (por ejemplo, el proceso de remuestreo), y una parte restante del conjunto de filtros no se puede utilizar para la conversión de frecuencia de muestreo (por ejemplo, no se puede utilizar para remuestreo). Algunos de estos ejemplos funcionan cuando se selecciona el índice de filtro actual utilizando un elemento de sintaxis que indica que la selección del índice de filtro actual de la parte restante del conjunto de filtros está prohibida.

5 [0111] El método 500 puede repetirse luego según sea necesario como parte de las operaciones de decodificación o codificación. En algunas operaciones, el método 500 no se utilizará para algunas iteraciones (por ejemplo, para la interpolación donde no está involucrado ningún remuestreo u otro proceso de conversión de frecuencia de muestreo, como cuando la imagen de referencia es del mismo tamaño que la imagen actual). Como se señaló con anterioridad, el proceso de conversión de la frecuencia de muestreo incluye un proceso de remuestreo en algunos ejemplos. Las operaciones del método 500 pueden incluir además identificar un primer grupo de filtros para interpolación y un 10 segundo grupo de filtros para remuestreo, en donde el índice de filtro actual es del segundo grupo de filtros. Las operaciones del método 500 pueden incluir además obtener una segunda imagen actual y una segunda imagen de referencia, determinando que un tercer valor de tamaño de imagen de la segunda imagen actual y un cuarto valor de tamaño de imagen de la segunda imagen de referencia tienen el mismo valor de tamaño. Basándose en la determinación de que el tercer valor de tamaño de imagen de la segunda imagen actual y el cuarto valor de tamaño de imagen de la segunda imagen de referencia tienen el mismo valor de tamaño, las operaciones del método 500 pueden incluir realizar un segundo proceso de remuestreo para un segundo bloque actual usando un segundo filtro seleccionado, en donde el segundo filtro seleccionado es del 15 primer grupo de filtros. En algunos de estos ejemplos, cada filtro del primer grupo de filtros y el segundo grupo de filtros se identifica mediante un índice de filtro asociado, el índice de filtro asociado indica un valor de índice asociado de un conjunto de filtros para interpolación o un conjunto de filtros para remuestreo. Algunos de estos ejemplos incluyen además obtener el índice de filtro actual a partir de un índice de filtro mediante 20 una operación de búsqueda para un valor de índice de filtro seleccionado, en donde el valor de índice de filtro seleccionado está asociado con el segundo grupo de filtros, y donde la asociación entre el valor de índice de filtro seleccionado y el segundo grupo de filtros se basa en determinar que el primer valor de tamaño de imagen y el segundo valor de tamaño de imagen son diferentes.

35 [0112] En algunos ejemplos, no se utiliza un índice de filtro actual (por ejemplo, se

ignora o un proceso de remuestreo u otro proceso de conversión de frecuencia de muestreo no depende del valor del índice de filtro actual) en el proceso de remuestreo cuando el ancho de imagen actual es diferente del ancho de la imagen de referencia o la altura de la imagen actual es diferente de la altura de la imagen de referencia. El

5 índice de filtro derivado o señalado se puede almacenar con el bloque actual y se utiliza en el proceso de derivación del candidato de vector de movimiento en los siguientes bloques. Cuando se determina que no se utiliza el índice de filtro derivado o señalado, el valor predeterminado para el índice de filtro se puede asignar temporalmente en el

10 proceso de remuestreo, mientras que el valor del índice de filtro derivado o señalado se mantiene sin cambios después del proceso de remuestreo. En este caso, el valor predeterminado puede indicar que no se aplica el filtro conmutable alternativo.

[0113] En algunos ejemplos ilustrativos, el índice de filtro se deriva de bloques vecinos durante el proceso de derivación de la lista de candidatos de vectores de movimiento para el modo de fusión, como se describió con anterioridad. De acuerdo con las

15 reivindicaciones adjuntas, el índice de filtro se señala a través del modo AMVR. En algunos ejemplos, un proceso de conversión de frecuencia de muestreo es un proceso de compensación de movimiento. En algunos de estos ejemplos, una imagen de referencia es una imagen de referencia utilizada para la interpretación del bloque actual de la imagen actual.

20 **[0114]** En algunos ejemplos, cuando se determina que no se usa el índice de filtro derivado o señalado, el valor del índice de filtro se puede reemplazar con un valor predeterminado. En algunos de estos ejemplos, el valor predeterminado indica que no se aplica el filtro conmutable alternativo, mientras que en un ejemplo un filtro alternativo puede ser un filtro de suavizado. En algunos otros ejemplos, cuando se determina que

25 el índice de filtro no se usa en el proceso de remuestreo (u otro proceso de conversión de frecuencia de muestreo), el valor del índice de filtro no se señala.

[0115] En algunas implementaciones, los procesos (o métodos) descritos en este documento se pueden realizar mediante un dispositivo informático o un aparato, como el sistema 100 que se muestra en la Figura 1. Por ejemplo, los procesos pueden ser

30 realizados por el dispositivo de codificación 104 mostrado en la Figura 1 y Figura 7, mediante otro dispositivo del lado de la fuente de video o dispositivo de transmisión de video, mediante el dispositivo de decodificación 112 mostrado en la Figura 1 y Figura 8, y/o por otro dispositivo del lado del cliente, como un dispositivo reproductor, una pantalla o cualquier otro dispositivo del lado del cliente. En algunos casos, el dispositivo o

35 aparato informático puede incluir un procesador, microprocesador, microordenador u

otro componente de un dispositivo que esté configurado para llevar a cabo los pasos de los procesos descritos en este documento. En algunos ejemplos, el dispositivo o aparato informático puede incluir una cámara configurada para capturar datos de video (por ejemplo, una secuencia de video) que incluyen cuadros de video. En algunos ejemplos, una cámara u otro dispositivo de captura que captura los datos de video está separado del dispositivo informático, en cuyo caso el dispositivo informático recibe u obtiene los datos de video capturados. El dispositivo informático puede incluir además una interfaz de red configurada para comunicar los datos de video. La interfaz de red puede configurarse para comunicar datos basados en el Protocolo de Internet (IP) u otro tipo de datos. En algunos ejemplos, el dispositivo o aparato informático puede incluir una pantalla para mostrar contenido de video de salida, como muestras de imágenes de un flujo de bits de video.

[0116] La Figura 6 es un diagrama de flujo que ilustra aspectos del remuestreo de imágenes de referencia de acuerdo con algunos ejemplos, mostrados en la Figura 6 como método 600. El método 600 se puede realizar como parte de un procedimiento de procesamiento de imagen o video, tal como un procedimiento de procesamiento de acuerdo con VVC, JEM u otros procedimientos estándar de codificación de video como se describe con anterioridad.

[0117] Durante el procesamiento de video como parte de una implementación del método 600, la operación 602 incluye identificar una imagen actual. Una imagen actual se puede identificar usando un POC como se describió con anterioridad, o usando cualquier otra operación para identificar los datos asociados con una imagen de un conjunto de imágenes (por ejemplo, un video) que se está procesando.

[0118] Para procesar la imagen actual identificada en la operación 602, se identifica una imagen de referencia en la operación 604. Como parte de esta identificación, se pueden seleccionar una o más imágenes de referencia o imágenes de referencia candidatas (por ejemplo, de un DPB) para una lista de imágenes de referencia (por ejemplo, usando un POC) como se describe en detalle anteriormente y a continuación. Dependiendo de la implementación, se puede seleccionar una imagen de referencia individual de la lista de imágenes de referencia para usar en el procesamiento de aspectos de la imagen actual. En algunas implementaciones, se pueden usar diferentes imágenes de referencia para procesar diferentes porciones de una imagen actual, pero durante el procesamiento, se emplean los datos de una imagen de referencia individual (por ejemplo, los datos de imagen usados para crear vectores de movimiento como el vector de movimiento 430 usando la imagen de referencia 410).

[0119] Además, durante el procesamiento de video de acuerdo con el método 600, se pueden usar varios modos diferentes como parte del procesamiento de video. Estos pueden incluir diferentes modos para la predicción de movimiento como se describe con anterioridad, como AMVR y modos de fusión. En diferentes implementaciones, estos
5 modos pueden cambiar durante el procesamiento de una imagen individual, partes de una imagen o grupos de imágenes.

[0120] Diferentes modos de procesamiento pueden señalar información para que los filtros se utilicen en el procesamiento de la imagen actual de diferentes maneras. Por ejemplo, como se describe en el presente documento, un AMVR puede señalar
10 explícitamente elementos de sintaxis que indican un valor de índice de filtro que identifica información sobre un filtro que se utilizará en el procesamiento de la imagen actual. Por el contrario, un modo de fusión puede funcionar sin señalización explícita del índice de filtro derivando la información del índice de filtro de otras imágenes (por ejemplo, imágenes adyacentes, imágenes de referencia, etc.). En la operación 608, se
15 determina la operación apropiada para identificar la información del índice de filtro, y la operación de procesamiento prosigue con las operaciones apropiadas para recopilar la información del índice de filtro. Si se usa señalización de índice de filtro explícito, entonces en la operación 610, los elementos de sintaxis de la señalización explícita se analizan para recopilar la información de índice de filtro (por ejemplo, un valor de un
20 índice de filtro). Si no se usa señalización de índice de filtro explícita, entonces en la operación 612, se deriva la información de índice de filtro (por ejemplo, se propaga desde una imagen o porción de imagen adyacente).

[0121] Una vez que se ha recopilado la información del índice del filtro, ya sea en la operación 610 o en la operación 612, la información del índice del filtro se usa para
25 establecer un valor de índice del filtro actual en la operación 614. Por ejemplo, en una implementación que usa la Tabla 1, el valor de índice de filtro se puede establecer en uno de los dos valores de índice de filtro asociados con la posición del índice fraccionaria de medio pel en función de la información del índice de filtro.

[0122] La operación 616 incluye una verificación de condición, que se usa para
30 seleccionar un filtro apropiado dependiendo de si se usa remuestreo. La verificación de condición implica comparar un tamaño de la imagen actual y la imagen de referencia identificada en la operación 604. Como se describe en este documento, esta verificación de condición puede implicar comparar un tamaño horizontal, un tamaño vertical, valores de resolución o cualquier otra información asociada con el tamaño de la imagen actual
35 y la imagen de referencia. En algunos ejemplos, la verificación de la condición (por

- ejemplo, comparar una referencia y el tamaño o escala de la imagen actual) se puede realizar por bloques. Por ejemplo, la verificación de condición se puede realizar para verificar si la imagen actual que contiene el bloque y una imagen de referencia indicada en el bloque por un índice de referencia tienen el mismo tamaño de imagen o si no tienen el mismo tamaño. En algunos ejemplos, la verificación de condición se puede realizar en un nivel de segmento, mosaico, subimagen u otro grupo de bloques. Por ejemplo, la verificación de condición puede verificar si alguna imagen de referencia disponible para la interpredicción y una imagen actual tienen el mismo tamaño.
- 5
- [0123]** En algunos ejemplos, la condición se puede verificar en un nivel de secuencia, por ejemplo, para verificar si es posible la posibilidad de tener diferentes tamaños de imagen. En un ejemplo, se puede indicar en un conjunto de parámetros (por ejemplo, mediante un indicador en un PPS, SPS y/o VPS) si se puede aplicar el remuestreo de imágenes.
- 10
- [0124]** En algunos casos, cuando se deriva el tamaño de la imagen (por ejemplo, para las imágenes actuales y de referencia), el tamaño puede ser un tamaño de imagen decodificado o un tamaño de imagen recortada. Por ejemplo, se puede señalar una ventana de conformidad en el flujo de bits y se puede aplicar mediante un decodificador o reproductor multimedia para recortar la imagen decodificada al tamaño de la imagen recortada. En algunos casos, el tamaño de la imagen para la verificación de la condición se puede derivar en función del tamaño de la imagen decodificada y varias compensaciones (por ejemplo, compensación izquierda, compensación derecha, compensación superior y/o compensación inferior). El tamaño de la imagen decodificada y/o las compensaciones se pueden señalar en un conjunto de parámetros (por ejemplo, PPS, SPS y/o VPS) o en otro lugar (por ejemplo, en un mensaje SEI, encabezado de segmento u otro mecanismo de señalización). En este caso, se puede derivar una relación de escala para el remuestreo considerando el tamaño de imagen de la imagen actual y el tamaño de una imagen de referencia con compensaciones aplicadas. En algunas implementaciones, las compensaciones pueden ser similares a las compensaciones de escala utilizadas en el estándar HEVC (por ejemplo, en la extensión escalable). Esta relación de escala y/o los valores de tamaño para la imagen actual y la imagen de referencia se pueden usar para determinar qué filtros se aplican al procesar la imagen actual. Las técnicas descritas en el presente documento pueden utilizar cualquier método para comprobar si un tamaño de imagen actual y un tamaño de imagen de referencia son del mismo tamaño.
- 15
- 20
- 25
- 30
- 35
- [0125]** Dependiendo del resultado de la operación 616, la imagen actual se puede

procesar de diferentes formas (por ejemplo, usando diferentes filtros). Si las imágenes actual y de referencia se identifican como del mismo tamaño, entonces en la operación 618, el valor del índice actual (por ejemplo, como se señaló o derivó con anterioridad) se usa en el procesamiento de la imagen actual.

- 5 **[0126]** Cuando la imagen actual y la imagen de referencia no tienen el mismo tamaño, en la operación 620, se ignora el valor del índice de filtro actual establecido en la operación 614. Esto puede implicar el uso de un valor de filtro predeterminado que se establece dentro del sistema para el filtrado que implica el remuestreo de la imagen de referencia, o un bucle de filtro predeterminado para operaciones donde esta condición
- 10 de verificación es verdadera. En algunos ejemplos, si se identifica que la condición de verificación es verdadera (por ejemplo, los tamaños de imagen de la imagen actual y una referencia, o una o ambas imágenes de referencia en la bipredicción son diferentes), entonces el uso de un filtro de interpolación conmutable se puede desactivar en un nivel de secuencia en SPS, un nivel de imagen a través de la señalización PPS, en un nivel
- 15 de segmento y/o mosaico, y/o en un nivel de bloque. Esta desactivación del filtrado de interpolación conmutable (por ejemplo, el uso del valor de índice de filtro actual establecido cuando se usa el filtrado de interpolación conmutable dentro de un sistema) puede considerarse como la selección del filtro predeterminado o cualquier filtro de reemplazo para el filtro de interpolación conmutable configurado en la operación 614.
- 20 **[0127]** Independientemente del filtro que se seleccione, la imagen actual se procesa usando el filtro seleccionado en la operación 622. Este procesamiento puede ser parte de un bucle de codificación que filtra diferentes porciones de la imagen actual y, por lo tanto, dependiendo de la implementación, las operaciones del método 600 puede repetirse y enlazarse de diferentes formas para procesar la imagen actual o partes de la
- 25 imagen actual usando diferentes filtros. En el ejemplo del método 600, la operación 624 incluye propagar el valor del índice de filtro actual de modo que esta información esté disponible para su uso con un bloque posterior si es necesario. Esto puede implicar la generación de señalización explícita, o puede implicar almacenar el valor de índice de filtro actual para que se pueda acceder a él durante una iteración posterior de la
- 30 operación 612 para derivar un valor de índice de filtro (por ejemplo, cuando la imagen actual o parte de la imagen actual se convierte en una imagen adyacente o de referencia para una imagen actual posterior que se está procesando). Entonces, el método 600 puede finalizar y se puede ejecutar en bucle para procesar una secuencia de video. Será evidente que el método 600 se puede implementar con operaciones adicionales
- 35 repetidas o intermedias, dependiendo de cómo se utilice el filtrado de interpolación

conmutable (por ejemplo, con estructuras para el filtrado de interpolación conmutable que se deshabilitarán en un nivel de secuencia en SPS, un nivel de imagen a través de la señalización PPS, a nivel de segmento y/o mosaico, y/o a nivel de bloque). Además, durante ciertos modos u operaciones de procesamiento, el filtrado de interpolación conmutable no se puede utilizar (por ejemplo, la selección de un cuarto de pel simple u otros valores distintos de los valores de índice de medio pel de la Tabla 1 con filtrado de interpolación conmutable), por lo que el método 600 no se puede usar para partes que procesan un video cuando se usa para otras partes del mismo video.

[0128] En algunos casos, cuando el uso de filtro de interpolación conmutable está deshabilitado en un nivel de bloque (por ejemplo, cuando se identifica que la condición de verificación es verdadera), un elemento de sintaxis que indica un índice de filtro (por ejemplo, señalización del modo AMVR de medio pel, en donde está implícitamente implícita una interpolación de medio pel diferente) no se señala para un bloque. En tales casos, los elementos de sintaxis asociados con el filtrado conmutable, por ejemplo, un índice de filtro, de AMVR de medio pel para indicar un filtro de interpolación diferente, están condicionados a la verificación del tamaño de la imagen (por ejemplo, si el tamaño de imagen actual y el tamaño de una imagen de referencia son del mismo tamaño).

[0129] En algunos casos, cuando el uso de filtro de interpolación conmutable está deshabilitado en un nivel de bloque (por ejemplo, cuando se identifica que la condición de verificación es verdadera), si un índice de filtro o el uso de un filtro diferente (por ejemplo, a través de, por ejemplo, señalización AMVR de medio pel o en un modo de fusión u otros modos similares a la fusión) se señala o se infiere de los bloques vecinos, el índice de filtro se modifica y se establece en un valor de índice de filtro predeterminado para el vector de movimiento derivado candidato de un bloque vecino. En un ejemplo, el filtro predeterminado puede indicar que no se aplica el filtro conmutable. De acuerdo con varios ejemplos, un filtro seleccionado puede indicarse explícita o implícitamente. En algunos ejemplos, un elemento de sintaxis explícito se señala mediante la señalización del modo AMVR. En otros ejemplos, el filtro está implícito, derivado o propagado desde un bloque vecino o bloque de referencia. Cuando el índice de filtro señalado (por ejemplo, explícita o implícitamente) no se usa debido a que el filtrado de interpolación conmutable está deshabilitado, se puede usar un valor de índice de filtro predeterminado. Este filtro predeterminado puede indicar que no se utiliza el filtrado conmutable (por ejemplo, como parte del proceso para deshabilitar el filtrado de interpolación conmutable cuando no está permitido).

[0130] En algunos casos, cuando el uso de filtro de interpolación conmutable está

deshabilitado en un nivel de bloque (cuando se identifica que la condición de verificación es verdadera), si un índice de filtro (o el uso de un filtro diferente a través, por ejemplo, señalización AMVR de medio pel) se infiere de los bloques vecinos, por ejemplo, en un modo de fusión u otros modos de fusión, el índice de filtro (o la señalización implícita a través, por ejemplo, del modo AMVR de medio pel) se deriva de bloques vecinos, pero durante el proceso de remuestreo, el índice de filtro derivado se ignora y se aplica un filtro de remuestreo incluso si puede indicar que se debe usar un filtrado conmutable. El índice de filtro derivado para un bloque actual se puede almacenar con el bloque y se puede usar para bloques posteriores (después del bloque actual), en cuyo caso el índice de filtro se propaga.

[0131] En algunos ejemplos, ampliando la idea anterior en los casos en que se puede seleccionar un filtro de un conjunto de filtros (por ejemplo, los dos filtros de medio pel de la Tabla 1), solo se puede usar un subconjunto de filtros para el remuestreo, y otros filtros que no se pueden usar para el remuestreo están restringidos para ser elegidos (en un ejemplo, el elemento de sintaxis que indica que tal selección puede estar prohibida en función de la condición de que el tamaño de la imagen sea diferente) o dicha selección se ignora cuando se realiza el proceso de remuestreo.

[0132] En algunos ejemplos, se pueden proporcionar dos grupos o conjuntos de filtros, en donde un grupo de filtros se usa para la interpolación (por ejemplo, para las operaciones de procesamiento 622 usando el filtro seleccionado en la operación 618) y otro grupo de filtros se usa para remuestrear (por ejemplo, la operación de procesamiento 622 usando el filtro seleccionado en la operación 620). En tales ejemplos, si se indica la conmutación del filtro, el filtro se selecciona de un grupo en función de la condición de que los tamaños de imagen sean diferentes, es decir, se realiza un remuestreo. Por ejemplo, un índice de filtro puede indicar el uso de un filtro con ese índice de filtro de un conjunto de filtros (o grupo), en donde el conjunto (o grupo) se identifica en función de si solo se realiza interpolación o remuestreo para el bloque.

[0133] En un ejemplo más detallado, suponiendo que hay un conjunto de filtros para la interpolación (indicado como *interpolationSet*) y otro conjunto de filtros para remuestreo (indicado como *resamplingSet*). Un conjunto de filtros puede tener un filtro o más de un filtro (el tamaño del conjunto de filtros puede ser 1 o mayor que 1). Se puede seleccionar un índice de filtro con un valor de *filterIndex* para usarlo en un bloque. Un filtro, que se utiliza en el bloque, se obtiene a partir de un conjunto de filtros buscando un filtro con el índice *filterIndex*. El filtro puede provenir del conjunto *interpolationSet* o del conjunto *resamplingSet*, y el conjunto de filtros se identifica en función de la condición de que los

tamaños de una imagen actual y una imagen de referencia sean iguales. Si la imagen actual y los tamaños de la imagen de referencia son los mismos (en cuyo caso no es necesario volver a muestrear), entonces el conjunto de filtros se identifica como *interpolationSet*. Si la imagen actual y los tamaños de la imagen de referencia son diferentes (en cuyo caso se realiza el remuestreo), el conjunto de filtros se identifica como *resamplingSet*. Una ventaja de esta técnica es que no hay necesidad de cambiar la señalización del índice del filtro y la inferencia del índice del filtro (por ejemplo, el valor del índice del filtro puede propagarse entre bloques vecinos interpolados o remuestreados), y solo el filtro final se selecciona de un conjunto basado en la condición del tamaño de la imagen.

[0134] Se describirá ahora un ejemplo ilustrativo de un proceso de RPR de remuestreo (o escalado) de imagen de referencia. Como se señaló con anterioridad, cuando hay un cambio de resolución (por ejemplo, dentro de una secuencia de video codificada (CVS)), una imagen puede tener un tamaño diferente de una o más de sus imágenes de referencia. El proceso de RPR se puede realizar para normalizar todos los vectores de movimiento a la cuadrícula de imágenes actual en lugar de sus correspondientes cuadrículas de imágenes de referencia. Este proceso puede resultar beneficioso para mantener la coherencia del diseño y hacer que los cambios de resolución sean transparentes para el proceso de predicción del vector de movimiento. De lo contrario, es posible que los vectores de movimiento vecinos que apuntan a imágenes de referencia con diferentes tamaños no se utilicen directamente para la predicción de vectores de movimiento espacial debido a la escala diferente.

[0135] Cuando ocurre un cambio de resolución, tanto los vectores de movimiento como los bloques de referencia se pueden escalar mientras se realiza una predicción con compensación de movimiento. En algunos casos, el rango de escala está limitado a $[1/8, 2]$ (por ejemplo, la escala ascendente está limitada a 1:8 y la escala descendente está limitada a 2:1). Aquí, aumento de escala (o sobremuestreo) se refiere al caso en donde la imagen de referencia es más pequeña que la imagen actual, mientras que la reducción de escala (o submuestreo) se refiere al caso en donde la imagen de referencia es más grande que la imagen actual. El proceso de escalado se describe con más detalle a continuación.

[0136] Para los bloques de luminancia, los factores de escala y sus representaciones de punto fijo se pueden definir como:

$$\text{hori\_scale\_fp} = \frac{(\text{width}_{ref} \ll 14) + (\frac{\text{width}_{cur}}{2})}{\text{width}_{cur}}, \text{ Ecuación (1)}$$

$$\text{vert\_scale\_fp} = \frac{(\text{height}_{ref} \ll 14) + (\frac{\text{height}_{cur}}{2})}{\text{height}_{cur}} \text{ Ecuación (2).}$$

[0137] El proceso de RPR puede incluir dos partes, que incluyen: (1) mapear el píxel de la esquina superior izquierda del bloque actual a la imagen de referencia; y (2) usar los tamaños de paso horizontal y vertical para abordar las ubicaciones de referencia de los otros píxeles del bloque actual. En algunos casos, si la coordenada del píxel de la esquina superior izquierda del bloque actual es (x, y), la ubicación de subpel (x', y') en la imagen de referencia apuntada por un vector de movimiento (mvX, mvY) en unidades de 1/16 de pel se especifica cuando la ubicación horizontal en la imagen de referencia es:

$$x' = ((x \ll 4) + \text{mvX}) \cdot \text{hori\_scale\_fp}, \text{ Ecuación (3)}$$

[0138] y x' se reduce aún más para mantener solo 10 bits fraccionarios:

$$x' = \text{Sign}(x') \cdot ((\text{Abs}(x') + (1 \ll 7)) \gg 8) \text{ Ecuación (4)}$$

[0139] De manera similar, la ubicación vertical en la imagen de referencia es:

$$y' = ((y \ll 4) + \text{mvY}) \cdot \text{vert\_scale\_fp}, \text{ Ecuación (5)}$$

[0140] e y' se reduce aún más a:

$$y' = \text{Sign}(y') \cdot ((\text{Abs}(y') + (1 \ll 7)) \gg 8) \text{ Ecuación (6)}$$

[0141] En este punto, la ubicación de referencia del píxel de la esquina superior izquierda del bloque actual está en (x', y'). Las otras ubicaciones de referencia subpel/pel se pueden calcular en relación con (x', y') con tamaños de paso horizontal y vertical. Esos tamaños de paso se pueden derivar con una precisión de 1/1024 pel (u otra precisión de pel) a partir de los factores de escala horizontal y vertical anteriores de la siguiente manera:

$$x\_step = (\text{hori\_scale\_fp} + 8) \gg 4, \text{ Ecuación (7)}$$

$$y\_step = (\text{vert\_scale\_fp} + 8) \gg 4 \text{ Ecuación (8)}$$

[0142] En un ejemplo, si un píxel en el bloque actual está en columna i y fila j de distancia del píxel de la esquina superior izquierda (o muestra), se derivan las coordenadas horizontales y verticales del píxel de referencia correspondiente por:

$$x'_i = x' + i * x\_step, \text{ Ecuación (9)}$$

$$y'_j = y' + j * y\_step \text{ Ecuación (10)}$$

[0143] En la interpolación subpel, es posible que sea necesario dividir x'_i e y'_j en partes de pel completo y partes de pel fraccionarias de la siguiente manera, en donde las partes de pel completo para direccionar el bloque de referencia son iguales a:

$$(x'_i + 32) \gg 10, \text{ Ecuación (11)}$$

$$(y'_j + 32) \gg 10 \text{ Ecuación (12)}$$

[0144] Las partes de pel fraccionarias utilizadas para seleccionar filtros de interpolación son iguales a

$$\Delta x = ((x'_i + 32) \gg 6) \& 15, \text{ Ecuación (13)}$$

$$5 \quad \Delta y = ((y'_j + 32) \gg 6) \& 15 \text{ Ecuación (14)}$$

[0145] Una vez que se determinan las ubicaciones de pel completa y de pel fraccionada dentro de una imagen de referencia, los interpoladores de compensación de movimiento existentes pueden usarse sin ningún cambio adicional. La ubicación del pel completo se utilizará para obtener el parche del bloque de referencia de la imagen de referencia y la
10 ubicación del pel fraccionaria se utilizará para seleccionar el filtro de interpolación adecuado.

[0146] Se describe ahora un ejemplo de un proceso RPR para bloques de crominancia. Por ejemplo, cuando el formato de crominancia es 4: 2: 0, los vectores de movimiento de crominancia tienen una precisión de 1/32 pel. En este ejemplo, el proceso de
15 escalado de los vectores de movimiento de crominancia y los bloques de referencia de crominancia es similar al de los bloques de luminancia, excepto que se introduce un ajuste relacionado con el formato de crominancia.

[0147] Cuando la coordenada del píxel de la esquina superior izquierda del bloque de crominancia actual es (x_c, y_c) , las ubicaciones horizontal y vertical iniciales en la imagen
20 de crominancia de referencia son

$$x'_c = ((x_c \ll 5) + mvX) \cdot \text{hori\_scale\_fp}, \text{ Ecuación (15)}$$

$$y'_c = ((y_c \ll 5) + mvY) \cdot \text{vert\_scale\_fp}, \text{ Ecuación (16)}$$

[0148] donde mvX y mvY son el vector de movimiento de luminancia original, pero ahora deben examinarse con una precisión de 1/32 pel. x'_c e y'_c se reducen aún más para
25 mantener una precisión de 1/1024 pel de la siguiente manera:

$$x'_c = \text{Sign}(x'_c) \cdot ((\text{Abs}(x'_c) + (1 \ll 8)) \gg 9), \text{ Ecuación (17)}$$

$$y'_c = \text{Sign}(y'_c) \cdot ((\text{Abs}(y'_c) + (1 \ll 8)) \gg 9) \text{ Ecuación (18)}$$

[0149] En comparación con las ecuaciones de luminancia asociadas, el desplazamiento a la derecha anterior se incrementa en un bit adicional.

[0150] Los tamaños de paso usados pueden ser los mismos que para luminancia. Para
30 un píxel de crominancia en (i, j) en relación con el píxel de la esquina superior izquierda, las coordenadas horizontales y verticales de su píxel de referencia se derivan de la siguiente manera:

$$x'_{c,i} = x'_c + i \cdot x\_step, \text{ Ecuación (19)}$$

$$y_{c'j} = y_{c'} + j * y\_step \text{ Ecuación (20)}$$

[0151] En la interpolación subpel, $x_{c'i}$ e $y_{c'j}$ también se dividen en partes de pel completas y partes de pel fraccionarias, y las partes de pel completas para direccionar el bloque de referencia son iguales a:

5 $(x_{c'i} + 16) \gg 10$, Ecuación (21)

$(y_{c'j} + 16) \gg 10$ Ecuación (22)

[0152] Las partes de pel fraccionarias utilizadas para seleccionar filtros de interpolación son iguales a:

$$\Delta x = ((x_{c'i} + 16) \gg 5) \& 31, \text{ Ecuación (23)}$$

10 $\Delta y = ((y_{c'j} + 16) \gg 5) \& 31$ Ecuación (24)

[0153] Los procesos se pueden describir con respecto a diagramas de flujo lógicos, cuya operación representa una secuencia de operaciones que pueden implementarse en hardware, instrucciones de computadora o una de sus combinaciones. En el contexto de las instrucciones de la computadora, las operaciones representan instrucciones ejecutables por computadora almacenadas en uno o más medios de almacenamiento legibles por computadora que, cuando son ejecutadas por uno o más procesadores, realizan las operaciones recitadas. En general, las instrucciones ejecutables por computadora incluyen rutinas, programas, objetos, componentes, estructuras de datos y similares que realizan funciones particulares o implementan tipos de datos particulares. El orden en que se describen las operaciones no debe interpretarse como una limitación, y cualquier número de las operaciones descritas se puede combinar en cualquier orden y/o en paralelo para implementar los procesos.

[0154] Además, los procesos pueden realizarse bajo el control de uno o más sistemas informáticos configurados con instrucciones ejecutables y pueden implementarse como código (por ejemplo, instrucciones ejecutables, uno o más programas informáticos, o una o más aplicaciones) ejecutándose colectivamente en uno o más procesadores, por hardware, o combinaciones de ellos. Como se indicó con anterioridad, el código puede almacenarse en un medio de almacenamiento legible por computadora o legible por máquina, por ejemplo, en forma de un programa informático que comprende una pluralidad de instrucciones ejecutables por uno o más procesadores. El medio de almacenamiento legible por computadora o por máquina puede no ser transitorio.

[0155] Las técnicas de codificación discutidas aquí pueden implementarse en un sistema de codificación y decodificación de video de ejemplo (por ejemplo, el sistema 100). En algunos ejemplos, un sistema incluye un dispositivo de origen que proporciona

datos de video codificados para ser decodificados posteriormente por un dispositivo de destino. En particular, el dispositivo de origen proporciona los datos de video al dispositivo de destino a través de un medio legible por computadora. El dispositivo de origen y el dispositivo de destino pueden comprender cualquiera de una amplia gama de dispositivos, incluidas computadoras de escritorio, computadoras notebook (es decir, computadoras portátiles), tabletas, decodificadores, terminales telefónicas como los llamados teléfonos “inteligentes”, o los llamados pads “inteligentes”, televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión de video o similares. En algunos casos, el dispositivo de origen y el dispositivo de destino pueden estar equipados para la comunicación inalámbrica.

[0156] El dispositivo de destino puede recibir los datos de video codificados para ser decodificados a través del medio legible por computadora. El medio legible por computadora puede comprender cualquier tipo de medio o dispositivo capaz de mover los datos de video codificados del dispositivo de origen al dispositivo de destino. En un ejemplo, el medio legible por computadora puede comprender un medio de comunicación para permitir que el dispositivo de origen transmita datos de video codificados directamente al dispositivo de destino en tiempo real. Los datos de video codificados pueden modularse de acuerdo con un estándar de comunicación, como un protocolo de comunicación inalámbrico, y transmitirse al dispositivo de destino. El medio de comunicación puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, como una red de área local, una red de área amplia o una red global como Internet. El medio de comunicación puede incluir enrutadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo de origen al dispositivo de destino.

[0157] En algunos ejemplos, los datos codificados pueden enviarse desde la interfaz de salida a un dispositivo de almacenamiento. De manera similar, se puede acceder a los datos codificados desde el dispositivo de almacenamiento mediante la interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera de una variedad de medios de almacenamiento de datos distribuidos o de acceso local, 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 otro ejemplo, el dispositivo de almacenamiento puede corresponder a

un servidor de archivos u otro dispositivo de almacenamiento intermedio que puede almacenar el video codificado generado por el dispositivo de origen. El dispositivo de destino puede acceder a los datos de video almacenados desde el dispositivo de almacenamiento mediante transmisión o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de video codificados y transmitir esos datos de video codificados al dispositivo de destino. Los servidores de archivos de ejemplo incluyen un servidor web (por ejemplo, para un sitio web), un servidor FTP, dispositivos de almacenamiento conectado a la red (NAS) o una unidad de disco local. El dispositivo de destino puede acceder a los datos de video codificados a través de cualquier conexión de datos estándar, incluida una conexión a Internet. Este dispositivo de destino puede incluir un canal inalámbrico (por ejemplo, una conexión Wi-Fi), una conexión por cable (por ejemplo, DSL, módem de cable, 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 desde el dispositivo de almacenamiento puede ser una transmisión de transmisión, una transmisión de descarga o una combinación de ellas.

[0158] En un ejemplo, el dispositivo de origen incluye una fuente de video, un codificador de video y una interfaz de salida. El dispositivo de destino puede incluir una interfaz de entrada, un decodificador de video y un dispositivo de visualización. El codificador de video del dispositivo de origen puede configurarse para aplicar las técnicas aquí descritas. En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen puede recibir datos de video de una fuente de video externa, como una cámara externa. Del mismo modo, el dispositivo de destino puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0159] El sistema de ejemplo anterior es simplemente un ejemplo. Las técnicas para procesar datos de video en paralelo pueden realizarse mediante cualquier dispositivo digital de codificación y/o decodificación de video. Aunque generalmente las técnicas de la presente divulgación se realizan mediante un dispositivo de codificación de video, las técnicas también se pueden realizar mediante un codificador/decodificador de video, típicamente denominado "CODEC". Además, las técnicas de la presente divulgación también pueden ser realizadas por un preprocesador de video. El dispositivo de origen y el dispositivo de destino son simplemente ejemplos de tales dispositivos de codificación en los que el dispositivo de origen genera datos de video codificados para su transmisión al dispositivo de destino. En algunos ejemplos, los dispositivos de origen

y de destino pueden funcionar de manera sustancialmente simétrica de manera que cada uno de los dispositivos incluya componentes de codificación y decodificación de video. Por lo tanto, los sistemas de ejemplo pueden admitir la transmisión de video unidireccional o bidireccional entre dispositivos de video, por ejemplo, para transmisión de video, reproducción de video, transmisión de video o telefonía de video.

[0160] La fuente de video puede incluir un dispositivo de captura de video, tal como una cámara de video, un archivo de video que contiene video previamente capturado, y/o una interfaz de alimentación de video para recibir video de un proveedor de contenido de video. Como alternativa adicional, la fuente de video puede generar datos basados en gráficos de computadora como el video fuente, o una combinación de video en vivo, video archivado y video generado por computadora. En algunos casos, si la fuente de video es una cámara de video, el dispositivo de origen y el dispositivo de destino pueden formar los llamados teléfonos con cámara o teléfonos con video. Sin embargo, como se mencionó con anterioridad, las técnicas descritas en la presente divulgación pueden ser aplicables a la codificación de video en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el video capturado, precapturado o generado por computadora puede ser codificado por el codificador de video. La información de video codificada se puede enviar luego por medio de una interfaz de salida al medio legible por computadora.

[0161] Como se señaló, el medio legible por computadora puede incluir medios transitorios, tales como una transmisión inalámbrica o transmisión de red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorios), como un disco duro, unidad flash, disco compacto, disco de video digital, disco Blu-ray u otros medios legibles por computadora. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de video codificados del dispositivo de origen y proporcionar los datos de video codificados al dispositivo de destino, por ejemplo, a través de la transmisión de red. De manera similar, un dispositivo informático de una instalación de producción media, como una instalación de estampado de disco, puede recibir datos de video codificados desde el dispositivo de origen y producir un disco que contenga los datos de video codificados. Por lo tanto, puede entenderse que el medio legible por computadora incluye uno o más medios legibles por computadora de diversas formas, en varios ejemplos.

[0162] La interfaz de entrada del dispositivo de destino recibe información del medio legible por computadora. La información del medio legible por computadora puede incluir información de sintaxis definida por el codificador de video, que también es utilizado por

el decodificador de video, que incluye elementos de sintaxis que describen características y/o procesamiento de bloques y otras unidades codificadas, por ejemplo, grupo de imágenes (GOP). Un dispositivo de visualización muestra los datos de video decodificados a un usuario, y puede comprender cualquiera de una variedad de dispositivos de visualización, como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, un diodo orgánico emisor de luz (OLED) u otro tipo de dispositivo de visualización. Se han descrito varias formas de realización de la aplicación.

[0163] Los detalles específicos del dispositivo de codificación 104 y el dispositivo de decodificación 112 se muestran en la Figura 7 y la Figura 8, respectivamente. La Figura 7 es un diagrama de bloques que ilustra un dispositivo 104 de codificación de ejemplo que puede implementar una o más de las técnicas descritas en la presente divulgación. El dispositivo de codificación 104 puede generar, por ejemplo, las estructuras de sintaxis descritas en este documento (por ejemplo, las estructuras de sintaxis de un VPS, SPS, PPS u otros elementos de sintaxis). El dispositivo de codificación 104 puede realizar codificación intrapredicción e interpredicción de bloques de video dentro de segmentos del video. Como se describió con anterioridad, la intracodificación se basa, al menos en parte, en la predicción espacial para reducir o eliminar la redundancia espacial dentro de un cuadro o imagen de video dado. La intercodificación se basa, al menos en parte, en la predicción temporal para reducir o eliminar la redundancia temporal dentro de los cuadros adyacentes o circundantes de una secuencia de video. Intramodo (modo I) puede referirse a cualquiera de varios modos de compresión basados en el espacio. Los intermodos, como la predicción unidireccional (modo P) o la bipredicción (modo B), pueden referirse a cualquiera de varios modos de compresión basados en el tiempo.

[0164] El dispositivo de codificación 104 incluye una unidad de partición 35, unidad de procesamiento de predicción 41, unidad de filtro 63, memoria de imágenes 64, sumador 50, unidad de procesamiento de transformada 52, unidad de cuantificación 54 y unidad de codificación de entropía 56. La unidad de procesamiento de predicción 41 incluye unidad de estimación de movimiento 42, unidad de compensación de movimiento 44 y unidad de procesamiento de intrapredicción 46. Para la reconstrucción de bloques de video, el dispositivo de codificación 104 también incluye la unidad de cuantificación inversa 58, la unidad de procesamiento de transformada inversa 60 y el sumador 62. La unidad de filtro 63 está destinada a representar uno o más filtros de bucle, como un filtro de desbloqueo, un filtro de bucle adaptativo (ALF) y un filtro de desplazamiento adaptativo de muestra (SAO). Aunque la unidad de filtro 63 se muestra en la Figura 7

como un filtro de bucle, en otras configuraciones, la unidad de filtro 63 puede implementarse como un filtro de bucle posterior. Un dispositivo de procesamiento posterior 57 puede realizar un procesamiento adicional en datos de video codificados generados por el dispositivo de codificación 104. Las técnicas de la presente divulgación pueden implementarse en algunos casos mediante el dispositivo de codificación 104. Sin embargo, en otros casos, una o más de las técnicas de la presente divulgación puede implementarse mediante el dispositivo de procesamiento posterior 57.

[0165] Como se muestra en la Figura 7, el dispositivo de codificación 104 recibe datos de video, y la unidad de partición 35 divide los datos en bloques de video. La partición también puede incluir la división en segmentos, porciones de segmentos, mosaicos u otras unidades más grandes, así como la división de bloques de video, por ejemplo, de acuerdo con una estructura quadtree de LCU y CU. El dispositivo de codificación 104 generalmente ilustra los componentes que codifican bloques de video dentro de un segmento de video a codificar. El segmento puede dividirse en múltiples bloques de video (y posiblemente en conjuntos de bloques de video conocidos como mosaicos). La unidad de procesamiento de predicción 41 puede seleccionar uno de una pluralidad de modos de codificación posibles, como uno de una pluralidad de modos de codificación de intrapredicción o uno de una pluralidad de modos de codificación de interpredicción, para el bloque de video actual basado en resultados de error (por ejemplo, tasa de codificación y el nivel de distorsión, o similares). La unidad de procesamiento de predicción 41 puede proporcionar el bloque intra- o intercodificado resultante al sumador 50 para generar datos de bloque residual y al sumador 62 para reconstruir el bloque codificado para su uso como imagen de referencia.

[0166] La unidad de procesamiento de intrapredicción 46 dentro de la unidad de procesamiento de predicción 41 puede realizar la codificación de intrapredicción del bloque de video actual en relación con uno o más bloques vecinos en el mismo cuadro o segmento que el bloque actual a codificar para proporcionar compresión espacial. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 dentro de la unidad de procesamiento de predicción 41 realizan una codificación interpredictiva del bloque de video actual en relación con uno o más bloques predictivos en una o más imágenes de referencia para proporcionar compresión temporal.

[0167] La unidad de estimación de movimiento 42 puede configurarse para determinar el modo de intrapredicción para un segmento de video de acuerdo con un patrón predeterminado para una secuencia de video. El patrón predeterminado puede designar segmentos de video en la secuencia como segmentos P, segmentos B o segmentos

GPB. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar altamente integradas, pero se ilustran por separado con fines conceptuales. La estimación de movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generar vectores de movimiento, que estiman el movimiento para bloques de video. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una unidad de predicción (PU) de un bloque de video dentro de un cuadro o imagen de video actual en relación con un bloque predictivo dentro de una imagen de referencia.

[0168] Un bloque predictivo es un bloque que se encuentra que coincide estrechamente con la PU del bloque de video a codificar en términos de diferencia de píxeles, que puede determinarse por la suma de la diferencia absoluta (SAD), la suma de la diferencia cuadrada (SSD) u otras métricas de diferencia. En algunos ejemplos, el dispositivo de codificación 104 puede calcular valores para posiciones de píxel subentero de imágenes de referencia almacenadas en la memoria de imágenes 64. Por ejemplo, el dispositivo de codificación 104 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones de píxeles fraccionarias de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento en relación con las posiciones de píxel completo y las posiciones de píxel fraccionarias y generar un vector de movimiento con precisión de píxel fraccionaria.

[0169] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de video en un segmento codificado comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia se puede seleccionar de una primera lista de imágenes de referencia (Lista 0) o una segunda lista de imágenes de referencia (Lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria de imágenes 64. La unidad de estimación de movimiento 42 envía el movimiento calculado a la unidad de codificación de vector a entropía 56 y a la unidad de compensación de movimiento 44.

[0170] La compensación de movimiento, realizada por la unidad de compensación de movimiento 44, puede implicar buscar o generar el bloque predictivo basado en el vector de movimiento determinado por la estimación de movimiento, posiblemente realizando interpolaciones a una precisión de subpíxel. Al recibir el vector de movimiento para la PU del bloque de video actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una lista de imágenes de referencia. El dispositivo de codificación 104 forma un bloque de video

residual restando los valores de píxeles del bloque predictivo de los valores de píxeles del bloque de video actual que se está codificando, formando valores de diferencia de píxeles. Los valores de diferencia de píxeles forman datos residuales para el bloque y pueden incluir componentes de diferencia de luminancia y crominancia. El sumador 50

5 representa el componente o componentes que realizan esta operación de resta. La unidad de compensación de movimiento 44 también puede generar elementos de sintaxis asociados con los bloques de video y el segmento de video para uso por el dispositivo de decodificación 112 en la decodificación de los bloques de video del segmento de video.

10 **[0171]** La unidad de procesamiento de intrapredicción 46 puede predecir dentro de un bloque actual, como una alternativa a la intrapredicción realizada por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se describió con anterioridad. En particular, la unidad de procesamiento de intrapredicción 46 puede determinar un modo de intrapredicción para usar para codificar un bloque

15 actual. En algunos ejemplos, la unidad de procesamiento de intrapredicción 46 puede codificar un bloque actual utilizando varios modos de intrapredicción (por ejemplo, durante pasadas de codificación separadas), y la unidad de procesamiento de intrapredicción 46 puede seleccionar un modo de intrapredicción apropiado para usar de modos probados. Por ejemplo, la unidad de procesamiento de intrapredicción 46

20 puede calcular los valores de distorsión de frecuencia utilizando un análisis de distorsión de frecuencia para los diversos modos de intrapredicción probados, y puede seleccionar el modo de intrapredicción que tenga las mejores características de distorsión de frecuencia entre los modos probados. El análisis de distorsión de tasa generalmente determina una cantidad de distorsión (o error) entre un bloque codificado y un bloque

25 original no codificado que se codificó para producir el bloque codificado, así como una tasa de bits (es decir, una cantidad de bits) utilizada para producir el bloque codificado. La unidad de procesamiento de intrapredicción 46 puede calcular relaciones a partir de las distorsiones y tasas para los diversos bloques codificados para determinar qué modo de intrapredicción exhibe el mejor valor de distorsión de tasa para el bloque.

30 **[0172]** En cualquier caso, después de seleccionar un modo de intrapredicción para un bloque, la unidad de procesamiento de intrapredicción 46 puede proporcionar información indicativa del modo de intrapredicción seleccionado para la unidad de codificación de bloque a entropía 56. La unidad de codificación de entropía 56 puede codificar la información que indica el modo intrapredicción seleccionado. El dispositivo

35 de codificación 104 puede incluir en la configuración de flujo de bits transmitido

definiciones de datos de contextos de codificación para varios bloques, así como indicaciones de un modo intrapredicción más probable, una tabla de índice de modo intrapredicción y una tabla de índice de modo intrapredicción modificada para utilizar para cada uno de los contextos. Los datos de configuración de flujo de bits pueden incluir una pluralidad de tablas de índice de modo de intrapredicción y una pluralidad de tablas de índice de modo de predicción modificada (también denominadas tablas de mapeo de palabras de código).

5 [0173] Después de que la unidad de procesamiento de predicción 41 genera el bloque predictivo para el bloque de video actual mediante interpredicción o intrapredicción, el dispositivo de codificación 104 forma un bloque de video residual restando el bloque predictivo del bloque de video actual. Los datos de video residual en el bloque residual pueden incluirse en una o más TU y aplicarse a la unidad de procesamiento de transformada 52. La unidad de procesamiento de transformada 52 transforma los datos de video residual en coeficientes de transformada residual usando una transformada, tal como una transformada de coseno discreto (DCT) o una transformada conceptualmente similar. La unidad de procesamiento de transformada 52 puede convertir los datos de video residuales de un dominio de píxeles a un dominio de transformada, tal como un dominio de frecuencia.

15 [0174] La unidad de procesamiento de transformada 52 puede enviar los coeficientes de transformada resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformada para reducir aún más la tasa de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos o todos los coeficientes. El grado de cuantificación puede modificarse ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar luego un escaneo de la matriz que incluye los coeficientes de transformada cuantificados. Alternativamente, la unidad de codificación de entropía 56 puede realizar el escaneo.

20 [0175] Después de la cuantificación, la unidad de codificación de entropía 56 codifica por entropía los coeficientes de transformada cuantificados. Por ejemplo, la unidad de codificación de entropía 56 puede realizar codificación de longitud variable adaptativa de contexto (CAVLC), codificación aritmética binaria adaptativa de contexto (CABAC), codificación aritmética binaria adaptativa de contexto basada en sintaxis (SBAC), codificación de entropía de división de intervalo de probabilidad (PIPE) u otra técnica de codificación de entropía. Después de la codificación de entropía por la unidad de codificación de entropía 56, el flujo de bits codificado puede transmitirse al dispositivo

25
30
35

de decodificación 112, o archivarse para su posterior transmisión o recuperación por el dispositivo de decodificación 112. La unidad de codificación de entropía 56 también puede codificar por entropía los vectores de movimiento y la otra sintaxis elementos para el segmento de video actual que se codifica.

- 5 **[0176]** La unidad de cuantificación inversa 58 y la unidad de procesamiento de transformada inversa 60 aplican cuantificación inversa y transformada inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles para su uso posterior como bloque de referencia de una imagen de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el
- 10 bloque residual a un bloque predictivo de una de las imágenes de referencia dentro de una lista de imágenes de referencia. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores de píxel subenteros para usar en la estimación de movimiento. El sumador 62 agrega el bloque residual reconstruido al bloque de predicción de
- 15 movimiento compensado producido por la unidad de compensación de movimiento 44 para producir un bloque de referencia para almacenamiento en la memoria de imágenes 64. El bloque de referencia puede ser utilizado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 como bloque de referencia para predecir un bloque en un cuadro de video o imagen posterior.
- 20 **[0177]** De esta manera, el dispositivo de codificación 104 de la Figura 7 representa un ejemplo de un codificador de video configurado para realizar cualquiera de las técnicas descritas en este documento, incluido el método 500 descrito con anterioridad con respecto a la Figura 5. En algunos casos, algunas de las técnicas de la presente divulgación también pueden implementarse mediante el dispositivo de procesamiento
- 25 posterior 57.

- [0178]** La Figura 8 es un diagrama de bloques que ilustra un ejemplo de dispositivo de decodificación 112. El dispositivo de decodificación 112 incluye una unidad de decodificación de entropía 80, unidad de procesamiento de predicción 81, unidad de cuantificación inversa 86, unidad de procesamiento de transformada inversa 88,
- 30 sumador 90, unidad de filtro 91 y memoria de imágenes 92. La unidad de procesamiento de predicción 81 incluye la unidad de compensación de movimiento 82 y la unidad de procesamiento de intrapredicción 84. El dispositivo de decodificación 112 puede realizar, en algunos ejemplos, una pasada de decodificación generalmente recíproca a la pasada de codificación descrita con respecto al dispositivo de codificación 104 de la
- 35 Figura 7.

[0179] Durante el proceso de decodificación, el dispositivo de decodificación 112 recibe un flujo de bits de video codificado que representa bloques de video de un segmento de video codificado y elementos de sintaxis asociados enviados por el dispositivo de codificación 104. En algunas formas de realización, el dispositivo de decodificación 112 puede recibir el flujo de bits de video codificado desde el dispositivo de codificación 104. En algunas formas de realización, el dispositivo de decodificación 112 puede recibir el flujo de bits de video codificado de una entidad de red 79, tal como un servidor, un elemento de red con reconocimiento de medios (MANE), un editor/empalmador de video u otro dicho dispositivo configurado para implementar una o más de las técnicas descritas con anterioridad. La entidad de red 79 puede incluir o no el dispositivo de codificación 104. Algunas de las técnicas descritas en la presente divulgación pueden implementarse por la entidad de red 79 antes de que la entidad de red 79 transmita el flujo de bits de video codificado al dispositivo de decodificación 112. En algunos sistemas de decodificación de video, la entidad de red 79 y el dispositivo de decodificación 112 pueden ser partes de dispositivos separados, mientras que en otros casos, la funcionalidad descrita con respecto a la entidad de red 79 puede ser realizada por el mismo dispositivo que comprende el dispositivo de decodificación 112.

[0180] La unidad de decodificación de entropía 80 del dispositivo de decodificación 112 decodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento y otros elementos de sintaxis. La unidad de decodificación de entropía 80 reenvía los vectores de movimiento y otros elementos de sintaxis a la unidad de procesamiento de predicción 81. El dispositivo de decodificación 112 puede recibir los elementos de sintaxis en el nivel de segmento de video y/o el nivel de bloque de video. La unidad de decodificación de entropía 80 puede procesar y analizar elementos de sintaxis de longitud fija y elementos de sintaxis de longitud variable en uno o más conjuntos de parámetros, tales como un VPS, SPS y PPS.

[0181] Cuando el segmento de video se codifica como un segmento intracodificado (I), la unidad de procesamiento de intrapredicción 84 de la unidad de procesamiento de predicción 81 puede generar datos de predicción para un bloque de video del segmento de video actual basado en un modo de intrapredicción señalado y datos de bloques previamente decodificados del cuadro o imagen actual. Cuando el cuadro de video se codifica como un segmento intercodificado (es decir, B, P o GPB), la unidad de compensación de movimiento 82 de la unidad de procesamiento de predicción 81 produce bloques predictivos para un bloque de video del segmento de video actual basado en los vectores de movimiento y otros elementos de sintaxis recibidos de la

unidad de decodificación de entropía 80. Los bloques predictivos pueden producirse a partir de una de las imágenes de referencia dentro de una lista de imágenes de referencia. El dispositivo de decodificación 112 puede construir las listas de cuadros de referencia, Lista 0 y Lista 1, usando técnicas de construcción por defecto basadas en imágenes de referencia almacenadas en la memoria de imágenes 92.

[0182] La unidad de compensación de movimiento 82 determina la información de predicción para un bloque de video del segmento de video actual analizando los vectores de movimiento y otros elementos de sintaxis, y usa la información de predicción para producir los bloques predictivos para el bloque de video actual que se decodifica.

Por ejemplo, la unidad de compensación de movimiento 82 puede usar uno o más elementos de sintaxis en un conjunto de parámetros para determinar un modo de predicción (por ejemplo, intra- o interpredicción) utilizado para codificar los bloques de video del segmento de video, un tipo de segmento de intrapredicción (por ejemplo, segmento B, segmento P o segmento GPB), información de construcción para una o más listas de imágenes de referencia para el segmento, vectores de movimiento para cada bloque de video intercodificado del segmento, estado de interpredicción para cada bloque de video codificado del segmento y otra información para decodificar los bloques de video en el segmento de video actual.

[0183] La unidad de compensación de movimiento 82 también puede realizar interpolación basada en filtros de interpolación. La unidad de compensación de movimiento 82 puede usar filtros de interpolación tal como los usa el dispositivo de codificación 104 durante la codificación de los bloques de video para calcular los valores interpolados para los píxeles subenteros de los bloques de referencia. En este caso, la unidad de compensación de movimiento 82 puede determinar los filtros de interpolación usados por el dispositivo de codificación 104 a partir de los elementos de sintaxis recibidos, y puede usar los filtros de interpolación para producir bloques predictivos.

[0184] La unidad de cuantificación inversa 86 cuantifica o descuantifica los coeficientes de transformada cuantificados proporcionados en el flujo de bits y decodificados por la unidad de decodificación de entropía 80. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el dispositivo de codificación 104 para cada bloque de video en el segmento de video para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que se debe aplicar. La unidad de procesamiento de transformada inversa 88 aplica una transformada inversa (por ejemplo, una DCT inversa u otra transformada inversa adecuada), una transformada entera inversa o un proceso de transformada inversa

conceptualmente similar, a los coeficientes de transformada para producir bloques residuales en el dominio de píxeles.

[0185] Después de que la unidad de compensación de movimiento 82 genera el bloque predictivo para el bloque de video actual basado en los vectores de movimiento y otros elementos de sintaxis, el dispositivo de decodificación 112 forma un bloque de video decodificado sumando los bloques residuales de la unidad de procesamiento de transformada inversa 88 con los bloques predictivos correspondientes generados por la unidad de compensación de movimiento 82. El sumador 90 representa el componente o componentes que realizan esta operación de suma. Si se desea, los filtros de bucle (ya sea en el bucle de codificación o después del bucle de codificación) también pueden usarse para suavizar las transiciones de píxeles o para mejorar la calidad del video. La unidad de filtro 91 está destinada a representar uno o más filtros de bucle tales como un filtro de desbloqueo, un filtro de bucle adaptativo (ALF) y un filtro de compensación adaptativa de muestra (SAO). Aunque la unidad de filtro 91 se muestra en la Figura 8 como un filtro de bucle interno, en otras configuraciones, la unidad de filtro 91 puede implementarse como un filtro de bucle posterior. Los bloques de video decodificados en un cuadro o imagen dados se almacenan luego en la memoria de imágenes 92, que almacena las imágenes de referencia utilizadas para la compensación de movimiento posterior. La memoria de imágenes 92 también almacena video decodificado para su posterior presentación en un dispositivo de visualización, tal como el dispositivo 122 de destino de video mostrado en la Figura 1.

[0186] De esta manera, el dispositivo de decodificación 112 de la Figura 8 representa un ejemplo de un decodificador de video configurado para realizar cualquiera de las técnicas descritas en este documento, incluido el método 300 descrito con anterioridad con respecto a la Figura 3.

[0187] Las técnicas de la presente divulgación no se limitan necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de video en apoyo de cualquiera de una variedad de aplicaciones multimedia, como transmisiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de video por Internet, como transmisión dinámica adaptativa a través de 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. En algunos ejemplos, el sistema puede configurarse para admitir transmisión de video unidireccional o bidireccional para admitir aplicaciones tales como transmisión de video, reproducción

de video, transmisión de video y/o telefonía de video.

[0188] Como se usa en el presente documento, la expresión “medio legible por computadora” incluye, pero no se limita a dispositivos de almacenamiento portátiles o no portátiles, dispositivos de almacenamiento óptico y varios otros medios capaces de almacenar, contener o transportar instrucciones y/o datos. Un medio legible por computadora puede incluir un medio no transitorio en donde se pueden almacenar datos y que no incluye ondas portadoras y/o señales electrónicas transitorias que se propagan de forma inalámbrica o por conexiones cableadas. Los ejemplos de un medio no transitorio pueden incluir, pero no se limita a un disco magnético o cinta, medios de almacenamiento óptico como un disco compacto (CD) o un disco versátil digital (DVD), memoria flash, memoria o dispositivos de memoria. Un medio legible por computadora puede haber almacenado en él código y/o instrucciones ejecutables por máquina que pueden representar un procedimiento, una función, un subprograma, un programa, una rutina, una subrutina, un módulo, un paquete de software, una clase o cualquier combinación de instrucciones, estructuras de datos o declaraciones de programa. Un segmento de código puede estar acoplado a otro segmento de código o un circuito de hardware pasando y/o recibiendo información, datos, argumentos, parámetros o contenidos de memoria. La información, los argumentos, los parámetros, los datos, etc. se pueden pasar, reenviar o transmitir a través de cualquier medio adecuado, incluido el intercambio de memoria, el paso de mensajes, el paso de token, la transmisión de red o similares.

[0189] En algunas formas de realización, los dispositivos de almacenamiento legibles por computadora, medios y memorias pueden incluir un cable o señal inalámbrica que contiene un flujo de bits y similares. Sin embargo, cuando se menciona, los medios de almacenamiento legibles por computadora no transitorios excluyen expresamente medios tales como energía, señales portadoras, ondas electromagnéticas y señales per se.

[0190] Se proporcionan detalles específicos en la descripción anterior para proporcionar una comprensión profunda de las formas de realización y ejemplos proporcionados en el presente documento. Sin embargo, un experto en la técnica entenderá que las formas de realización se pueden practicar sin estos detalles específicos. Para mayor claridad de la explicación, en algunos casos, la presente tecnología puede presentarse como que incluye bloques funcionales individuales que incluyen bloques funcionales que comprenden dispositivos, componentes de dispositivos, pasos o rutinas en un método incorporado en software o combinaciones de hardware y software. Se pueden usar

componentes adicionales distintos a los mostrados en las figuras y/o descritos en el presente documento. Por ejemplo, los circuitos, sistemas, redes, procesos y otros componentes pueden mostrarse como componentes en forma de diagrama de bloques para no oscurecer las formas de realización con detalles innecesarios. En otros casos, pueden mostrarse circuitos, procesos, algoritmos, estructuras y técnicas bien conocidos sin detalles innecesarios para evitar oscurecer las formas de realización.

[0191] Las formas de realización individuales se pueden describir con anterioridad como un proceso o método que se representa como un flujograma, un diagrama de flujo, un diagrama de flujo de datos, un diagrama de estructura o un diagrama de bloques.

Aunque un diagrama de flujo puede describir las operaciones como un proceso secuencial, muchas de las operaciones se pueden realizar en paralelo o simultáneamente. Además, el orden de las operaciones puede reorganizarse. Un proceso finaliza cuando se completan sus operaciones, pero podría tener pasos adicionales no incluidos en una figura. Un proceso puede corresponder a un método, una función, un procedimiento, una subrutina, un subprograma, etc. Cuando un proceso corresponde a una función, su terminación puede corresponder a un retorno de la función a la función de llamada o la función principal.

[0192] Los procesos y métodos de acuerdo con los ejemplos descritos con anterioridad pueden implementarse usando instrucciones ejecutables por computadora que se almacenan o están disponibles de otro modo desde medios legibles por computadora. Dichas instrucciones pueden incluir, por ejemplo, instrucciones y datos que causan o configuran una computadora de propósito general, una computadora de propósito especial o un dispositivo de procesamiento para realizar una determinada función o grupo de funciones. Se puede acceder a partes de los recursos informáticos utilizados en una red. Las instrucciones ejecutables de la computadora pueden ser, por ejemplo, binarios, instrucciones de formato intermedio tales como lenguaje ensamblador, firmware, código fuente, etc. Ejemplos de medios legibles por computadora que pueden usarse para almacenar instrucciones, información utilizada y/o información creada durante los métodos de acuerdo con los ejemplos descritos incluyen discos magnéticos u ópticos, memoria flash, dispositivos USB provistos de memoria no volátil, dispositivos de almacenamiento en red, etc.

[0193] Los dispositivos que implementan procesos y métodos de acuerdo con estas divulgaciones pueden incluir hardware, software, firmware, middleware, microcódigo, lenguajes de descripción de hardware, o cualquier combinación de ellos, y pueden tomar cualquiera de una variedad de factores de forma. Cuando se implementa en software,

firmware, middleware o microcódigo, el código del programa o los segmentos de código para realizar las tareas necesarias (por ejemplo, un producto de programa de computadora) pueden almacenarse en un medio legible por computadora o legible por máquina. Un procesador o procesadores pueden realizar las tareas necesarias. Los
5 ejemplos típicos de factores de forma incluyen computadoras portátiles, teléfonos inteligentes, teléfonos móviles, tabletas u otras computadoras personales de factor de tamaño pequeño, asistentes digitales personales, dispositivos de montaje en bastidor, dispositivos independientes, etc. La funcionalidad descrita en este documento también puede realizarse en periféricos o tarjetas adicionales. Dicha funcionalidad también se
10 puede implementar en una placa de circuito entre diferentes chips o diferentes procesos que se ejecutan en un solo dispositivo, a modo de ejemplo adicional.

[0194] Las instrucciones, los medios para transmitir tales instrucciones, los recursos informáticos para ejecutarlos y otras estructuras para soportar tales recursos informáticos son medios de ejemplo para proporcionar las funciones descritas en la
15 divulgación.

[0195] En la descripción anterior, los aspectos de la solicitud se describen con referencia a formas de realización específicas de la misma, pero los expertos en la técnica reconocerán que la solicitud no se limita a los mismos. Por lo tanto, si bien las formas de realización ilustrativas de la solicitud se han descrito en detalle en este
20 documento, debe entenderse que los conceptos de la invención pueden de otra manera incorporarse y emplearse de manera diversa, y que las reivindicaciones adjuntas pretenden interpretarse para incluir tales variaciones, excepto según las limitaciones por el estado de la técnica. Varias características y aspectos de la solicitud descrita con anterioridad pueden usarse individual o conjuntamente. Por consiguiente, la memoria
25 descriptiva y los dibujos deben considerarse ilustrativos más que restrictivos. Para fines de ilustración, los métodos se describieron en un orden particular. Debería apreciarse que, en formas de realización alternativas, los métodos pueden realizarse en un orden diferente al descrito.

[0196] Un experto habitual apreciará que los símbolos o la terminología menor que (" $<$ ") y mayor que (" $>$ ") utilizados en este documento pueden reemplazarse por menor o igual que (" $\leq$ ") y mayor o igual que (" $\geq$ "), respectivamente, sin apartarse del alcance de esta descripción.
30

[0197] Cuando los componentes se describen como "configurados para" realizar ciertas operaciones, dicha configuración se puede lograr, por ejemplo, mediante el diseño de
35 circuitos electrónicos u otro hardware para realizar la operación, mediante la

programación de circuitos electrónicos programables (por ejemplo, microprocesadores u otros circuitos electrónicos adecuados)) para realizar la operación, o cualquier combinación de ellas.

5 **[0198]** La frase “acoplado a” se refiere a cualquier componente que esté físicamente conectado a otro componente, ya sea directa o indirectamente, y/o cualquier componente que esté en comunicación con otro componente (por ejemplo, conectado al otro componente a través de una conexión por cable o inalámbrica, y/u otra interfaz de comunicación adecuada) directa o indirectamente.

10 **[0199]** El lenguaje de acuerdo con la reivindicación u otro lenguaje que recite “al menos uno de” un conjunto y/o “uno o más” de un conjunto indica que un miembro del conjunto o varios miembros del conjunto (en cualquier combinación) satisfacen la reivindicación. Por ejemplo, la frase de las reivindicaciones que recite “al menos uno de A y B” o “al menos uno de A o B” significa A, B o A y B. En otro ejemplo, la frase de las reivindicaciones que recite “al menos uno de A, B, y C” o “al menos uno de A, B o C”
15 significa A, B, C, o A y B, o A y C, o B y C, o A y B y C. La frase “en al menos uno de” un conjunto y/o “uno o más” de un conjunto no limita el conjunto a los elementos enumerados en el conjunto. Por ejemplo, la frase de las reivindicaciones que recita “al menos uno de A y B” o “al menos uno de A o B” puede significar A, B o A y B, y además puede incluir elementos que no figuran en el conjunto de A y B.

20 **[0200]** Los diversos bloques lógicos, módulos, circuitos y pasos de algoritmo ilustrativos descritos en relación con las formas de realización descritas en este documento pueden implementarse como hardware electrónico, software, firmware o combinaciones de ellos. Para ilustrar claramente esta intercambiabilidad de hardware y software, varios componentes, bloques, módulos, circuitos y pasos ilustrativos se han descrito con
25 anterioridad generalmente en términos de su funcionalidad. Si dicha funcionalidad se implementa como hardware o software depende de la aplicación particular y las restricciones de diseño impuestas en el sistema general. Los expertos en la técnica pueden implementar la funcionalidad descrita de diferentes maneras para cada aplicación en particular, pero tales decisiones de implementación no deben interpretarse
30 como una desviación del alcance de la presente solicitud.

[0201] Las técnicas descritas en el presente documento también pueden implementarse en hardware electrónico, software, firmware o cualquier combinación de ellos. Dichas técnicas pueden implementarse en cualquiera de una variedad de dispositivos, tales como computadoras de uso general, dispositivos inalámbricos de comunicación o
35 dispositivos de circuitos integrados que tienen múltiples usos, incluida la aplicación en

dispositivos inalámbricos de comunicación y otros dispositivos. Todas las características descritas como módulos o componentes pueden implementarse juntas en un dispositivo lógico integrado o por separado como dispositivos lógicos discretos pero interoperables. Si se implementa en software, las técnicas se pueden realizar al menos en parte

5 mediante un medio de almacenamiento de datos legible por computadora que comprende un código de programa que incluye instrucciones que, cuando se ejecutan, realizan uno o más de los métodos descritos con anterioridad. El medio de almacenamiento de datos legible por computadora puede formar parte de un producto de programa de computadora, que puede incluir materiales de embalaje. El medio

10 legible por computadora puede comprender memoria o medios de almacenamiento de datos, tales como memoria de acceso aleatorio (RAM) tal como memoria de acceso aleatorio dinámico síncrono (SDRAM), memoria de solo lectura (ROM), memoria de acceso aleatorio no volátil (NVRAM), memoria de solo lectura programable y borrrable eléctricamente (EEPROM), memoria FLASH, medios de almacenamiento de datos

15 ópticos o magnéticos, y similares. Las técnicas, adicional o alternativamente, pueden realizarse al menos en parte mediante un medio de comunicación legible por computadora que transporta o comunica el código del programa en forma de instrucciones o estructuras de datos y que una computadora puede acceder, leer y/o ejecutar, como señales propagadas u ondas.

20 **[0202]** El código del programa puede ser ejecutado por un procesador, que puede incluir uno o más procesadores, tales como uno o más procesadores de señal digital (DSP), microprocesadores de propósito general, circuitos integrados específicos de una aplicación (ASIC), puertas lógicas programables en campo. (FPGA) u otro circuito lógico integrado o discreto equivalente. Dicho procesador puede configurarse para realizar

25 cualquiera de las técnicas descritas en la presente divulgación. Un procesador de propósito general puede ser un microprocesador; pero, como alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador o máquina de estado convencional. Un procesador también puede implementarse como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un

30 microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores junto con un núcleo DSP o cualquier otra configuración de este tipo. Por consiguiente, el término “procesador”, como se usa en el presente documento, puede referirse a cualquiera de la estructura anterior, cualquier combinación de la estructura anterior o cualquier otra estructura o aparato adecuado para la implementación de las técnicas

35 descritas en el presente documento. Además, en algunos aspectos, la funcionalidad

descrita en este documento puede proporcionarse dentro de módulos de software dedicados o módulos de hardware configurados para codificar y decodificar, o incorporarse en un codificador-decodificador de video combinado (CODEC).