

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 996 341**

51 Int. Cl.:

<b>H04N 19/103</b>	(2014.01)
<b>H04N 19/105</b>	(2014.01)
<b>H04N 19/109</b>	(2014.01)
<b>H04N 19/174</b>	(2014.01)
<b>H04N 19/176</b>	(2014.01)
<b>H04N 19/436</b>	(2014.01)
<b>H04N 19/52</b>	(2014.01)
<b>H04N 19/96</b>	(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **12.08.2019 PCT/CN2019/100226**
- 87 Fecha y número de publicación internacional: **13.02.2020 WO20030187**
- 96 Fecha de presentación y número de la solicitud europea: **12.08.2019 E 19846142 (8)**
- 97 Fecha y número de publicación de la concesión europea: **09.10.2024 EP 3815358**

54 Título: **Método de procesamiento de video, aparato de procesamiento de video, codificador, decodificador, medio y programa informático**

30 Prioridad:  
**10.08.2018 US 201862717004 P**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:  
**12.02.2025**

73 Titular/es:  
**HUAWEI TECHNOLOGIES CO., LTD. (100.00%)  
Huawei Administration Building Bantian,  
Longgang District  
Shenzhen, Guangdong 518129, CN**

72 Inventor/es:  
**KOTRA, ANAND, MEHER;  
ESENLIK, SEMIH;  
CHEN, JIANLE;  
WANG, BIAO;  
GAO, HAN y  
ZHAO, ZHIJIE**

74 Agente/Representante:  
**PONS ARIÑO, Ángel**

ES 2 996 341 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

**DESCRIPCIÓN**

Método de procesamiento de vídeo, aparato de procesamiento de vídeo, codificador, decodificador, medio y programa informático

5

Campo técnico

Las realizaciones de la presente solicitud (descripción) generalmente se refieren al campo de la codificación de vídeo y más particularmente a un método de procesamiento de vídeo, un aparato de procesamiento de vídeo, un codificador, un decodificador, un medio y un programa informático.

10

Antecedentes

La codificación de vídeo (codificación y decodificación de vídeo) se usa en una amplia gama de aplicaciones de vídeo digital, por ejemplo, TV digital de radiodifusión, transmisión de vídeo a través de Internet y redes móviles, aplicaciones conversacionales en tiempo real tales como videoconferencias, DVD y discos Blu-ray, sistemas de adquisición y edición de contenido de vídeo, y videocámaras de aplicaciones de seguridad.

15

Desde el desarrollo del enfoque de codificación de vídeo híbrido basado en bloques en el estándar H.261 en 1990, se desarrollaron nuevas técnicas y herramientas de codificación de vídeo y se formaron las bases para nuevos estándares de codificación de vídeo. Otros estándares de codificación de vídeo comprenden vídeo MPEG-1, vídeo MPEG-2, H.262/MPEG-2 de ITU-T, H.263 de ITU-T, H.264/Parte 10 de MPEG-4 de ITU-T, codificación de vídeo avanzada (AVC), H.265 de ITU-T/codificación de vídeo de alta eficiencia (HEVC), H.266 de ITU-T/codificación de vídeo versátil (VVC) y extensiones, por ejemplo, escalabilidad y/o extensiones tridimensionales (3D), de estos estándares. A medida que la creación y el uso de vídeo se han vuelto más y más ubicuos, el tráfico de vídeo es la mayor carga en las redes de comunicación y almacenamiento de datos, en consecuencia, uno de los objetivos de la mayoría de los estándares de codificación de vídeo fue lograr una reducción de la tasa de bits en comparación con su predecesor sin sacrificar la calidad de la imagen. Incluso la última codificación de vídeo de alta eficiencia (HEVC) puede comprimir el vídeo aproximadamente el doble que el AVC sin sacrificar la calidad, y hay una necesidad de comprimir aún más los vídeos en comparación con el HEVC.

20

25

30

El documento JVET-K0104 describe un método de predicción de vectores de movimiento basado en el historial para la intercodificación.

35

La solicitud de patente de EE. UU. US 2017/0127086 A1 se refiere a un método y aparato de codificación de vídeo para un bloque de datos de profundidad o datos de textura codificados en SSM (modo de muestra única). Las realizaciones según la invención construyen una lista de candidatos de muestra que comprende uno o más candidatos de color único que corresponden a una o más muestras representativas de uno o más bloques codificados con SSM anteriores, o uno o más candidatos de color de paleta que corresponden a uno o más colores usados anteriormente en una o más paletas anteriores asociadas con uno o más bloques codificados por paleta, o ambos. A continuación, se determina un candidato de muestra seleccionado a partir de la lista de candidatos de muestra y el candidato de muestra seleccionado se usa para codificar o decodificar el bloque actual representando todo el bloque actual mediante este candidato de muestra seleccionado. Los candidatos espaciales y/o temporales de bloques anteriormente codificados con SSM también se pueden incluir en la lista de candidatos de muestra.

40

45

Compendio

Las realizaciones de la presente solicitud proporcionan un método de procesamiento de vídeo y un aparato correspondiente, para mejorar la eficiencia de codificación.

50

El objeto de las reivindicaciones independientes permite alcanzar los objetivos anteriores y otros. Otras formas de implementación se desprenden de las reivindicaciones dependientes, de la descripción y de las figuras.

55

Un primer aspecto de la presente invención proporciona un método de procesamiento de vídeo, caracterizado por que comprende: al comienzo del procesamiento de cada fila de unidad de árbol de codificación, CTU, de un área de imagen que consiste en una pluralidad de filas CTU, inicializar una lista de predicción de vector de movimiento basado en el historial, HMVP, para cada una de la pluralidad de filas CTU, en donde las listas HMVP inicializadas están vacías, en donde cada fila CTU se mantiene con una lista HMVP diferente; procesar la pluralidad de filas CTU basándose en las listas HMVP, en donde una fila CTU actual de la pluralidad de filas CTU se procesa basándose en una lista HMVP inicializada para la fila CTU actual.

60

Se puede ver que la lista HMVP para la fila CTU actual se inicializa al comienzo para procesar la fila CTU actual, el proceso de la fila CTU actual no necesita basarse en una lista HMVP de una fila CTU anterior y, por lo tanto, puede mejorar la eficiencia de codificación y la eficiencia de decodificación.

65

Con referencia al primer aspecto, en una primera manera de implementación posible del primer aspecto, la cantidad de vectores de movimiento candidatos en la lista HMVP inicializada es cero.

5 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una segunda manera de implementación posible del primer aspecto, la fila CTU actual pertenece a un área de imagen que consiste en una pluralidad de filas CTU, y la fila CTU actual es una cualquiera de la pluralidad de filas CTU, por ejemplo, la primera fila CTU (por ejemplo, superior), la segunda fila CTU, ... y la última fila CTU (por ejemplo, inferior) del área de imagen.

10 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una tercera manera de implementación posible del primer aspecto, el método comprende además: inicializar una lista HMVP para cada una de la pluralidad de filas CTU excepto la fila CTU actual, en donde las listas HMVP para la pluralidad de filas CTU son idénticas o diferentes. En otras palabras, las realizaciones pueden además inicializar listas HMVP para todas las demás filas CTU del área de imagen, es decir, pueden inicializar listas HMVP para todas las filas CTU del área de imagen.

15 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una cuarta forma de implementación posible del primer aspecto, el procesamiento de la fila de CTU actual basándose en la lista HMVP comprende: procesar la CTU actual de la fila de CTU actual; actualizar la lista HMVP inicializada basándose en la CTU actual procesada; y procesar la segunda CTU de la fila de CTU actual basándose en la lista HMVP actualizada.

20 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente, en una quinta manera de implementación posible del primer aspecto, la lista HMVP se actualiza según una CTU procesada de la fila CTU actual.

25 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una sexta manera de implementación posible del primer aspecto, la lista HMVP para la fila CTU actual se inicializa de la siguiente manera: vaciar la lista HMVP para la fila CTU actual.

30 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una séptima manera de implementación posible del primer aspecto, el procesamiento de la fila CTU actual basándose en la lista HMVP comprende: procesar la fila CTU actual basándose en la lista HMVP de la segunda CTU de la fila CTU actual, en donde la segunda CTU es adyacente a la CTU inicial.

35 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una octava manera de implementación posible del primer aspecto, la pluralidad de filas CTU se procesan en modo de procesamiento paralelo de frente de onda (WPP).

40 Se puede ver que, como la lista HMVP para la fila CTU actual se inicializa al comienzo para procesar la fila CTU actual, cuando se combina con el modo WPP, las filas CTU de un fotograma de imagen o un área de imagen se pueden procesar en paralelo y, por lo tanto, pueden mejorar aún más la eficiencia de codificación y la eficiencia de decodificación.

45 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una novena manera de implementación posible del primer aspecto, la fila de CTU actual comienza a procesarse (o el procesamiento de la fila de CTU actual comienza) cuando se procesa una CTU particular de una fila CTU anterior.

50 Con referencia al primer aspecto de o una cualquiera de las maneras de implementación mencionadas anteriormente del primer aspecto, en una décima manera de implementación posible del primer aspecto, la fila CTU anterior es la fila CTU inmediatamente adyacente a la fila CTU actual y encima de o por encima de la fila CTU actual.

55 Con referencia a la novena manera de aplicación del primer aspecto de o la décima manera de implementación del primer aspecto, en una undécima manera de implementación posible del primer aspecto, la CTU particular de la fila CTU anterior es la segunda CTU de la fila CTU anterior; o la CTU particular de la fila CTU anterior es la primera CTU de la fila CTU anterior.

60 Un segundo aspecto de la presente invención proporciona un soporte de almacenamiento legible por ordenador que almacena instrucciones informáticas, que cuando se ejecutan por uno o más procesadores, hacen que el uno o más procesadores realicen el método según el primer aspecto o una cualquiera de las maneras de implementación del primer aspecto.

65

Un tercer aspecto de la presente invención proporciona un decodificador, que comprende: uno o más procesadores; y un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para ser ejecutada por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el decodificador para llevar a cabo el método según una cualquiera del primer aspecto o una cualquiera de las maneras de implementación del primer aspecto.

Un cuarto aspecto de la presente invención proporciona un codificador, que comprende: uno o más procesadores; y un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para ser ejecutada por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el codificador para llevar a cabo el método según el primer aspecto o una cualquiera de las maneras de implementación del primer aspecto.

Breve descripción de los dibujos

En las siguientes realizaciones de la invención se describen más detalles con referencia a las figuras y dibujos adjuntos, en los que:

la FIG. 1A es un diagrama de bloques que muestra un ejemplo de un sistema de codificación de vídeo configurado para implementar realizaciones de la invención;

la FIG. 1B es un diagrama de bloques que muestra otro ejemplo de un sistema de codificación de vídeo configurado para implementar realizaciones de la invención;

la FIG. 2 es un diagrama de bloques que muestra un ejemplo de un sistema de codificación de vídeo configurado para implementar realizaciones de la invención;

la FIG. 3 es un diagrama de bloques que muestra una estructura de ejemplo de un decodificador de vídeo configurado para implementar realizaciones de la invención;

la FIG. 4 es un diagrama de bloques que ilustra un ejemplo de un aparato de codificación o un aparato de decodificación;

la FIG. 5 es un diagrama de bloques que ilustra otro ejemplo de aparato de codificación o un aparato de decodificación;

la FIG. 6 muestra las posiciones de los bloques vecinos espaciales usados en la construcción de la lista de candidatos a fusión y AMVP;

la FIG. 7 es un diagrama de flujo de decodificación de un método HMVP;

la FIG. 8 es un diagrama de bloques que ilustra una orden de procesamiento de WPP;

la FIG. 9 es un diagrama de flujo que ilustra una operación de ejemplo de un decodificador de vídeo según una realización;

la FIG. 10 es un diagrama de flujo que ilustra una operación de ejemplo de una realización;

la FIG. 11 es un diagrama de bloques que ilustra un ejemplo de un aparato de procesamiento de vídeo.

En los símbolos de referencia idénticos siguientes se hace referencia a características idénticas o, al menos, funcionalmente equivalentes si no hay ninguna nota específica sobre la diferencia de esos símbolos de referencia idénticos.

Descripción detallada de las realizaciones

En la siguiente descripción, se hace referencia a las figuras adjuntas, que forman parte de la descripción, y que muestran, a modo de ilustración, aspectos específicos de realizaciones de la invención o aspectos específicos en los que se pueden usar las realizaciones de la presente invención. Se entiende que las realizaciones de la invención se pueden usar en otros aspectos y comprenden cambios estructurales o lógicos no representados en las figuras. Por lo tanto, la siguiente descripción detallada no debe tomarse en un sentido limitado, y el alcance de la presente invención se define por las reivindicaciones adjuntas.

Por ejemplo, se entiende que una descripción en relación con un método descrito también puede ser válida para un dispositivo o sistema correspondiente configurado para realizar el método y viceversa. Por ejemplo, si se describen uno o una pluralidad de etapas del método específicas, un dispositivo correspondiente puede

5 incluir una o una pluralidad de unidades, por ejemplo, unidades funcionales, para realizar la una o la pluralidad de etapas del método descritas (por ejemplo, una unidad que realiza la una o la pluralidad de etapas, o una pluralidad de unidades, cada una de las cuales realiza una o más de la pluralidad de etapas), incluso si tales una o más unidades no se describen o ilustran explícitamente en las figuras. Por otro lado, por ejemplo, si un aparato específico se describe en función de una o una pluralidad de unidades, por ejemplo, unidades funcionales, un método correspondiente puede incluir una etapa para realizar la funcionalidad de la una o la pluralidad de unidades (por ejemplo, una etapa que realiza la funcionalidad de la una o la pluralidad de unidades, o una pluralidad de etapas, cada una de las cuales realiza la funcionalidad de una o más de la pluralidad de unidades), incluso si tal una o pluralidad de etapas no se describen o ilustran explícitamente en las figuras. Además, se entiende que las características de las diversas realizaciones ejemplares y/o los aspectos descritos en la presente memoria pueden combinarse entre sí, a menos que se indique específicamente lo contrario.

15 La codificación de vídeo generalmente se refiere al procesamiento de una secuencia de imágenes, que forman el vídeo o la secuencia de vídeo. En lugar del término "imagen", el término "fotograma" o "imagen" se pueden usar como sinónimos en el campo de la codificación de vídeo. La codificación de vídeo usada en la presente solicitud (o la presente descripción) indica codificación de vídeo o decodificación de vídeo. La codificación de vídeo se realiza en el lado de origen, lo que generalmente comprende el procesamiento (por ejemplo, mediante compresión) de las imágenes de vídeo originales para reducir la cantidad de datos necesarios para representar las imágenes de vídeo (para un almacenamiento y/o transmisión más eficiente). La decodificación de vídeo se realiza en el lado de destino y generalmente comprende el procesamiento inverso en comparación con el codificador para reconstruir las imágenes de vídeo. Las realizaciones que se refieren a la "codificación" de imágenes de vídeo (o imágenes en general, como se explicará más adelante) se entenderán relacionadas con la "codificación" o la "decodificación" de secuencias de vídeo. La combinación de la parte de codificación y la parte de decodificación también se denomina CÓDEC (codificación y decodificación).

20 En caso de codificación de vídeo sin pérdidas, las imágenes de vídeo originales pueden reconstruirse, es decir, las imágenes de vídeo reconstruidas tienen la misma calidad que las imágenes de vídeo originales (suponiendo que no haya pérdida de transmisión u otra pérdida de datos durante el almacenamiento o la transmisión). En caso de codificación de vídeo con pérdida, se realiza una compresión adicional, por ejemplo, por cuantificación, para reducir la cantidad de datos que representan las imágenes de vídeo, que no pueden reconstruirse completamente en el decodificador, es decir, la calidad de las imágenes de vídeo reconstruidas es menor o peor en comparación a la calidad de las imágenes de vídeo originales.

30 Varios estándares de codificación de vídeo desde H.261 pertenecen al grupo de "códecs de vídeo híbridos con pérdida" (es decir, combinan predicción espacial y temporal en el dominio de la muestra y codificación de transformación 2D para aplicar cuantificación en el dominio de transformación). Cada imagen de una secuencia de vídeo generalmente se divide en un conjunto de bloques que no se superponen y la codificación generalmente se realiza a nivel de bloque. En otras palabras, en el codificador, el vídeo generalmente se procesa, es decir, se codifica, en un nivel de bloque (bloque de vídeo), por ejemplo, usando predicción espacial (intraimagen) y predicción temporal (interimagen) para generar un bloque de predicción, restando el bloque de predicción del bloque actual (bloque actualmente procesado/a procesar) para obtener un bloque residual, transformando el bloque residual y cuantificando el bloque residual en el dominio de transformación para reducir la cantidad de datos a transmitir (compresión), mientras que en el decodificador se aplica parcialmente al bloque codificado o comprimido el procesamiento inverso al del codificador para reconstruir el bloque actual para su representación. Además, el codificador duplica el bucle de procesamiento del decodificador de manera que ambos generarán predicciones idénticas (por ejemplo, intrapredicciones e interpredicciones) y/o reconstrucciones para el procesamiento, es decir, la codificación, de los bloques posteriores.

40 Como se usa en la presente memoria, el término "bloque" puede ser una parte de una imagen o un fotograma. Para facilitar la descripción, las realizaciones de la invención se describen en la presente memoria en referencia a la codificación de vídeo de alta eficiencia (HEVC) o al software de referencia de codificación de vídeo versátil (VVC), desarrollado por el equipo de colaboración conjunta en codificación de vídeo (JCT-VC) del grupo de expertos en codificación de vídeo (VCEG) del ITU-T y del grupo de expertos en imágenes en movimiento (MPEG) del ISO/IEC. Un experto en la técnica comprenderá que las realizaciones de la invención no se limitan a HEVC o VVC. Puede referirse a una CU, PU y TU. En HEVC, una CTU se divide en CU usando una estructura de árbol cuádruple denominada árbol de codificación. La decisión de codificar un área de imagen usando predicción interimagen (temporal) o intraimagen (espacial) se toma en el nivel de CU. Cada CU se puede dividir en una, dos o cuatro PU según el tipo de división de PU. Dentro de una PU, se aplica el mismo proceso de predicción y la información relevante se transmite al decodificador en función de la PU. Después de obtener el bloque residual aplicando el proceso de predicción basándose en el tipo de división de PU, una CU se puede dividir en unidades de transformación (TU) según otra estructura de árbol cuádruple similar al árbol de codificación de la CU. En el desarrollo más reciente de la técnica de compresión de vídeo, el fotograma de particionamiento de árbol cuádruple y árbol binario (QTBT) se usa para dividir un bloque de codificación. En la estructura de bloques QTBT, una CU puede tener una forma cuadrada o rectangular. Por ejemplo, una unidad de árbol de codificación (CTU) se divide primero por una estructura de árbol cuádruple. Los nodos de hoja de

5 árbol cuádruple se dividen además por una estructura de árbol binario. Los nodos de hoja del árbol binario se denominan unidades de codificación (CU), y esa segmentación se usa para el procesamiento de predicción y transformación sin ningún otro particionamiento. Esto significa que la CU, PU y TU tienen el mismo tamaño del bloque en la estructura de bloques de codificación QTBT. En paralelo, también se propuso usar la partición múltiple, por ejemplo, la partición de árbol triple junto con la estructura de bloques QTBT.

Se describen un codificador 20, un decodificador 30 y un sistema 10 de codificación en las siguientes realizaciones basándose en las FIGs. 1 a 3.

10 La FIG. 1A es un diagrama de bloques conceptual o esquemático que ilustra un sistema 10 de codificación de ejemplo, por ejemplo, un sistema 10 de codificación de vídeo que puede utilizar técnicas de esta presente solicitud (presente descripción). El codificador 20 (por ejemplo, el codificador 20 de vídeo) y el decodificador 30 (por ejemplo, el decodificador 30 de vídeo) del sistema 10 de codificación de vídeo representan ejemplos de dispositivos que pueden configurarse para realizar técnicas según diversos ejemplos descritos en la presente solicitud. Como se muestra en la FIG. 1A, el sistema 10 de codificación comprende un dispositivo 12 de origen configurado para proporcionar datos 13 codificados, por ejemplo, una imagen 13 codificada, por ejemplo, a un dispositivo 14 de destino para decodificar los datos 13 codificados.

20 El dispositivo 12 de origen comprende un codificador 20, y puede comprender adicionalmente, es decir, opcionalmente, un origen 16 de imágenes, una unidad 18 de preprocesamiento, por ejemplo, una unidad 18 de preprocesamiento de imágenes, y una interfaz de comunicación o unidad 22 de comunicación.

25 El origen 16 de imágenes puede comprender o ser cualquier tipo de dispositivo de captura de imágenes, por ejemplo, para capturar una imagen del mundo real, y/o cualquier tipo de dispositivo de generación de imágenes o comentarios (para la codificación del contenido de la pantalla, algunos textos en la pantalla también se consideran parte de una imagen o una imagen a codificar), por ejemplo, un procesador de gráficos por ordenador para generar una imagen animada por ordenador, o cualquier tipo de dispositivo para obtener y/o proporcionar una imagen del mundo real, una imagen animada por ordenador (por ejemplo, un contenido de pantalla, una imagen de realidad virtual (VR)) y/o cualquier combinación de las mismas (por ejemplo, una imagen de realidad aumentada (AR)).

30 Una imagen (digital) es o puede ser considerada como un conjunto o matriz bidimensional de muestras con valores de intensidad. Una muestra en la matriz también puede denominarse píxel (forma abreviada de elemento de imagen) o pel. El número de muestras en dirección horizontal y vertical (o eje) de la matriz o imagen define el tamaño y/o resolución de la imagen. Para la representación del color, generalmente se emplean tres componentes de color, es decir, la imagen puede representarse o incluir tres matrices de muestra. En espacio de color o formato RGB, una imagen comprende una matriz de muestra roja, verde y azul correspondiente. Sin embargo, en la codificación de vídeo, cada píxel generalmente se representa en un espacio de color o formato de luminancia/crominancia, por ejemplo, YCbCr, que comprende un componente de luminancia indicada por Y (a veces también se usa L en su lugar) y dos componentes de crominancia indicados por Cb y Cr. El componente Y de luminancia (o de forma abreviada, luma) representa el brillo o la intensidad del nivel de grises (por ejemplo, como en una imagen en escala de grises), mientras que los dos componentes Cb y Cr de crominancia (o de forma abreviada, croma) representan los componentes de cromaticidad o información de color. En consecuencia, una imagen en formato YCbCr comprende una matriz de muestras de valores de muestras de luminancia (Y) y dos matrices de muestras de crominancia de valores de crominancia (Cb y Cr). Las imágenes en formato RGB se pueden convertir o transformar en formato YCbCr y viceversa, el proceso también se conoce como transformación o conversión de color. Si una imagen es monocromática, la imagen puede comprender solo una matriz de muestras de luminancia.

50 En el muestreo monocromo solo hay una matriz de muestras, que nominalmente se considera la matriz de luma.

55 En el muestreo 4:2:0, cada una de las dos matrices de croma tiene la mitad de la altura y la mitad del ancho de la matriz de luma.

En el muestreo 4:2:2, cada una de las dos matrices de croma tiene la misma altura y la mitad del ancho de la matriz de luma.

60 En el muestreo 4:4:4, dependiendo del valor de `separate_color_plane_flag`, se aplica lo siguiente:

- Si `separate_color_plane_flag` es igual a 0, cada una de las dos matrices de croma tiene la misma altura y ancho que la matriz de luma;

65 - De lo contrario (`separate_colour_plane_flag` es igual a 1), los tres planos de color se procesan por separado como imágenes de muestra monocromas.

El origen 16 de imágenes (por ejemplo, el origen 16 de vídeo) puede ser, por ejemplo, una cámara para capturar una imagen, una memoria, por ejemplo, una memoria de imágenes, que comprende o almacena una imagen anteriormente capturada o generada, y/o cualquier tipo de interfaz (interna o externa) para obtener o recibir una imagen. La cámara puede ser, por ejemplo, una cámara local o una cámara integrada en el dispositivo de origen, la memoria puede ser una memoria local o integrada, por ejemplo, integrada en el dispositivo de origen. La interfaz puede ser, por ejemplo, una interfaz externa para recibir una imagen de un origen de vídeo externo, por ejemplo, un dispositivo externo de captura de imágenes tal como una cámara, una memoria externa o un dispositivo externo de generación de imágenes, por ejemplo, un procesador de gráficos externo, un ordenador o un servidor. La interfaz puede ser cualquier tipo de interfaz, por ejemplo, una interfaz por cable o inalámbrica, una interfaz óptica, según cualquier protocolo de interfaz patentado o estandarizado. La interfaz para obtener los datos 17 de imagen puede ser la misma interfaz o una parte de la interfaz 22 de comunicación.

A diferencia de la unidad 18 de preprocesamiento y el procesamiento realizado por la unidad 18 de preprocesamiento, la imagen o los datos 17 de imagen (por ejemplo, los datos 16 de vídeo) también pueden denominarse imagen sin procesar o datos 17 de imagen sin procesar.

La unidad 18 de preprocesamiento está configurada para recibir los datos 17 de imagen (sin procesar) y para realizar el preprocesamiento de los datos 17 de imagen para obtener una imagen 19 preprocesada o datos 19 de imagen preprocesados. El preprocesamiento realizado por la unidad 18 de preprocesamiento puede comprender, por ejemplo, recorte, conversión de formato de color (por ejemplo, de RGB a YCbCr), corrección de color o eliminación de ruido. Puede entenderse que la unidad 18 de preprocesamiento puede ser un componente opcional.

El codificador 20 (por ejemplo, el codificador 20 de vídeo) está configurado para recibir los datos 19 de imagen preprocesados y proporcionar datos 21 de imagen codificados (a continuación se describirán más detalles, por ejemplo, basándose en la FIG. 2 o la FIG. 4).

La interfaz 22 de comunicación del dispositivo 12 de origen puede configurarse para recibir los datos 21 de imagen codificados y transmitirlos a otro dispositivo, por ejemplo, el dispositivo 14 de destino o cualquier otro dispositivo, para almacenamiento o reconstrucción directa, o para procesar los datos 21 de imagen codificados respectivamente antes de almacenar los datos 13 codificados y/o transmitir los datos 13 codificados a otro dispositivo, por ejemplo, el dispositivo 14 de destino o cualquier otro dispositivo para decodificación o almacenamiento.

El dispositivo 14 de destino comprende un decodificador 30 (por ejemplo, un decodificador 30 de vídeo), y puede comprender adicionalmente, es decir, opcionalmente, una interfaz de comunicación o unidad 28 de comunicación, una unidad 32 de posprocesamiento y un dispositivo 34 de visualización.

La interfaz 28 de comunicación del dispositivo 14 de destino está configurada para recibir los datos 21 de imagen codificados o los datos 13 codificados, por ejemplo, directamente desde el dispositivo 12 de origen o desde cualquier otro origen, por ejemplo, un dispositivo de almacenamiento, por ejemplo, un dispositivo de almacenamiento de datos de imagen codificados.

La interfaz 22 de comunicación y la interfaz 28 de comunicación pueden configurarse para transmitir o recibir los datos 21 de imagen codificados o los datos 13 codificados a través de un enlace de comunicación directo entre el dispositivo 12 de origen y el dispositivo 14 de destino, por ejemplo, una conexión directa por cable o inalámbrica, o a través de cualquier tipo de red, por ejemplo, una red por cable o inalámbrica o cualquier combinación de las mismas, o cualquier tipo de red privada y pública, o cualquier tipo de combinación de las mismas.

La interfaz 22 de comunicación puede configurarse, por ejemplo, para empaquetar los datos 21 de imagen codificados en un formato apropiado, por ejemplo, paquetes, para la transmisión a través de un enlace de comunicación o una red de comunicación.

La interfaz 28 de comunicación, que forma la contraparte de la interfaz 22 de comunicación, puede configurarse, por ejemplo, para desempaquetar los datos 13 codificados para obtener los datos 21 de imagen codificados.

Tanto la interfaz 22 de comunicación como la interfaz 28 de comunicación pueden configurarse como interfaces de comunicación unidireccionales, como lo indica la flecha para los datos 13 de imagen codificados en la FIG. 1A que apuntan desde el dispositivo 12 de origen al dispositivo 14 de destino, o interfaces de comunicación bidireccionales, y puede configurarse, por ejemplo, para enviar y recibir mensajes, por ejemplo, para establecer una conexión, para acuse de recibo e intercambiar cualquier otra información relacionada con el enlace de comunicación y/o transmisión de datos, por ejemplo, transmisión de datos de imágenes codificados.

El decodificador 30 está configurado para recibir los datos 21 de imagen codificados y proporcionar datos 31

de imagen decodificados o una imagen 31 decodificada (a continuación se describirán más detalles, por ejemplo, basándose en la FIG. 3 o la FIG. 5).

5 El posprocesador 32 del dispositivo 14 de destino está configurado para posprocesar los datos 31 de imagen decodificados (también denominados datos de imagen reconstruidos), por ejemplo, la imagen 31 decodificada, para obtener datos 33 de imagen posprocesados, por ejemplo, una imagen 33 posprocesada. El posprocesamiento realizado por la unidad 32 de posprocesamiento puede comprender, por ejemplo, conversión de formato de color (por ejemplo, de YCbCr a RGB), corrección de color, recorte o remuestreo, o cualquier otro procesamiento, por ejemplo, para preparar los datos 31 de imagen decodificados para la visualización, por ejemplo, mediante el dispositivo 34 de visualización.

10 El dispositivo 34 de visualización del dispositivo 14 de destino está configurado para recibir los datos 33 de imagen posprocesados para visualizar la imagen, por ejemplo, a un usuario o espectador. El dispositivo 34 de visualización puede ser o comprender cualquier tipo de pantalla para representar la imagen reconstruida, por ejemplo, una pantalla o monitor integrado o externo. Las pantallas pueden comprender, por ejemplo, pantallas de cristal líquido (LCD), pantallas de diodos orgánicos emisores de luz (OLED), pantallas de plasma, proyectores, pantallas microLED, cristal líquido sobre silicio (LCoS), procesador de luz digital (DLP) o cualquier otro tipo de pantalla.

15 Aunque la FIG. 1A representa el dispositivo 12 de origen y el dispositivo 14 de destino como dispositivos separados, las realizaciones de los dispositivos también pueden comprender ambos o ambas funcionalidades, el dispositivo 12 de origen o la funcionalidad correspondiente y el dispositivo 14 de destino o la funcionalidad correspondiente. En tales realizaciones, el dispositivo 12 de origen o la funcionalidad correspondiente y el dispositivo 14 de destino o la funcionalidad correspondiente pueden implementarse usando el mismo hardware y/o software o por hardware y/o software separados o cualquier combinación de los mismos.

20 Como será evidente para el experto en la técnica basándose en la descripción, la existencia y división (exacta) de funcionalidades de las diferentes unidades o funcionalidades dentro del dispositivo 12 de origen y/o el dispositivo 14 de destino, como se muestra en la FIG. 1A, puede variar dependiendo del dispositivo y la aplicación reales.

25 El codificador 20 (por ejemplo, un codificador 20 de vídeo) y el decodificador 30 (por ejemplo, un decodificador 30 de vídeo) pueden implementarse cada uno como cualquiera de una variedad de circuitos adecuados, tal como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas programables en campo (FPGA), lógica discreta, hardware o cualquier combinación de los mismos. Si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un soporte de almacenamiento legible por ordenador no transitorio adecuado y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta descripción. Cualquiera de los anteriores (incluyendo hardware, software, una combinación de hardware y software, etc.) puede considerarse como uno o más procesadores. Cada uno del codificador 20 de vídeo y del decodificador 30 de vídeo puede incluirse en uno o más codificadores o decodificadores, cualquiera de los cuales puede integrarse como parte de un codificador/decodificador (CÓDEC) combinado en un dispositivo respectivo.

30 El dispositivo 12 de origen puede denominarse dispositivo de codificación de vídeo o aparato de codificación de vídeo. El dispositivo 14 de destino puede denominarse dispositivo de decodificación de vídeo o aparato de decodificación de vídeo. El dispositivo 12 de origen y el dispositivo 14 de destino pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo.

35 El dispositivo 12 de origen y el dispositivo 14 de destino pueden comprender cualquiera de una amplia gama de dispositivos, incluido cualquier tipo de dispositivo portátil o fijo, por ejemplo, ordenadores portátiles o notebook, teléfonos móviles, teléfonos inteligentes, tabletas u ordenadores tabletas, cámaras, ordenadores de escritorio, decodificadores, televisores, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión de vídeo (tales como servidores de servicios de contenido o servidores de entrega de contenido), dispositivo receptor de emisiones, dispositivo transmisor de emisiones o similares, y puede usar ningún o cualquier tipo de sistema operativo.

40 En algunos casos, el dispositivo 12 de origen y el dispositivo 14 de destino pueden estar equipados para comunicación inalámbrica. Por lo tanto, el dispositivo 12 de origen y el dispositivo 14 de destino pueden ser dispositivos de comunicación inalámbrica.

45 En algunos casos, el sistema 10 de codificación de vídeo ilustrado en la FIG. 1A es simplemente un ejemplo y las técnicas de la presente solicitud pueden aplicarse a configuraciones de codificación de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no necesariamente incluyen ninguna comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, los datos se recuperan de una memoria local, se transmiten a través de una red o similar. Un dispositivo de codificación de vídeo puede

codificar y almacenar datos en la memoria, y/o un dispositivo de decodificación de vídeo puede recuperar y decodificar datos de la memoria. En algunos ejemplos, la codificación y decodificación la realizan dispositivos que no se comunican entre sí, sino que simplemente codifican datos en la memoria y/o recuperan y decodifican datos de la memoria.

5

Debe entenderse que, para cada uno de los ejemplos anteriores descritos con referencia al codificador 20 de vídeo, el decodificador 30 de vídeo puede configurarse para realizar un proceso recíproco. Con respecto a los elementos de sintaxis de señalización, el decodificador 30 de vídeo puede configurarse para recibir y analizar tal elemento de sintaxis y decodificar los datos de vídeo asociados en consecuencia. En algunos ejemplos, el codificador 20 de vídeo puede codificar por entropía uno o más elementos de sintaxis en el flujo de bits de vídeo codificado. En tales ejemplos, el decodificador 30 de vídeo puede analizar tal elemento de sintaxis y decodificar los datos de vídeo asociados en consecuencia.

10

La FIG. 1B es un diagrama ilustrativo de otro ejemplo del sistema 40 de codificación de vídeo que incluye el codificador 20 de la FIG. 2 y/o el decodificador 30 de la FIG. 3 según una realización ejemplar. El sistema 40 puede implementar técnicas según diversos ejemplos descritos en la presente solicitud. En la implementación ilustrada, el sistema 40 de codificación de vídeo puede incluir dispositivo(s) 41 de generación de imágenes, codificador 100 de vídeo, decodificador 30 de vídeo (y/o un codificador de vídeo implementado a través del circuito 47 lógico de la(s) unidad(es) 46 de procesamiento), una antena 42, uno o más procesador(es) 43, uno o más almacenamiento(s) 44 de memoria, y/o un dispositivo 45 de visualización.

15

20

Como se ilustra, el(los) dispositivo(s) 41 de generación de imágenes, la antena 42, la(s) unidad(es) 46 de procesamiento, el circuito 47 lógico, el codificador 20 de vídeo, el decodificador 30 de vídeo, el(los) procesador(es) 43, el(los) almacenamiento(s) 44 de memoria, y/o el dispositivo 45 de visualización pueden ser capaces de comunicarse entre sí. Como se analizó, aunque se ilustra tanto con el codificador 20 de vídeo como con el decodificador 30 de vídeo, el sistema 40 de codificación de vídeo puede incluir solo el codificador 20 de vídeo o solo el decodificador 30 de vídeo en diversos ejemplos.

25

Como se muestra, en algunos ejemplos, el sistema 40 de codificación de vídeo puede incluir una antena 42. La antena 42 puede configurarse para transmitir o recibir un flujo de bits codificado de datos de vídeo, por ejemplo. Además, en algunos ejemplos, el sistema 40 de codificación de vídeo puede incluir un dispositivo 45 de visualización. El dispositivo 45 de visualización puede configurarse para presentar datos de vídeo. Como se muestra, en algunos ejemplos, el circuito 47 lógico puede implementarse a través de la(s) unidad(es) 46 de procesamiento. La(s) unidad(es) 46 de procesamiento pueden incluir lógica de circuito integrado de aplicación específica (ASIC), procesador(es) de gráficos, procesador(es) de propósito general, o similares. El sistema 40 de codificación de vídeo también puede incluir procesador(es) 43 opcional(es), que pueden incluir del mismo modo lógica de circuito integrado de aplicación específica (ASIC), procesador(es) de gráficos, procesador(es) de propósito general o similares. En algunos ejemplos, el circuito 47 lógico puede implementarse a través de hardware, hardware dedicado de codificación de vídeo o similar, y el(los) procesador(es) 43 pueden implementar software de propósito general, sistemas operativos o similares. Además, el(los) almacenamiento(s) 44 de memoria pueden ser cualquier tipo de memoria, tales como memoria volátil (por ejemplo, memoria de acceso aleatorio estática (SRAM), memoria de acceso aleatorio dinámica (DRAM), etc.) o memoria no volátil (por ejemplo, memoria flash, etc.), etc. En un ejemplo no limitativo, el(los) almacenamiento(s) 44 de memoria pueden implementarse mediante memoria caché. En algunos ejemplos, el circuito 47 lógico puede acceder al(a los) almacenamiento(s) 44 de memoria (para la implementación de un búfer de imágenes, por ejemplo). En otros ejemplos, el circuito 47 lógico y/o la(s) unidad(es) 46 de procesamiento pueden incluir almacenamientos de memoria (por ejemplo, memoria caché o similar) para la implementación de un búfer de imágenes o similar.

30

35

40

45

En algunos ejemplos, el codificador 100 de vídeo implementado a través de un circuito lógico puede incluir un búfer de imágenes (por ejemplo, a través de unidad(es) de 46 procesamiento o almacenamiento(s) 44 de memoria) y una unidad de procesamiento de gráficos (por ejemplo, a través de una(s) unidad(es) 46 de procesamiento). La unidad de procesamiento de gráficos puede estar acoplada comunicativamente al búfer de imágenes. La unidad de procesamiento de gráficos puede incluir un codificador 100 de vídeo implementado a través del circuito 47 lógico para incorporar los diversos módulos como se analizó con respecto a la FIG. 2 y/o cualquier otro sistema o subsistema codificador descrito en la presente memoria. El circuito lógico puede configurarse para realizar las diversas operaciones que se analizan en la presente memoria.

50

55

El decodificador 30 de vídeo puede implementarse de manera similar a como se implementa a través del circuito 47 lógico para incorporar los diversos módulos como se analizó con respecto al decodificador 30 de la FIG. 3 y/o cualquier otro sistema o subsistema decodificador descrito en la presente memoria. En algunos ejemplos, el decodificador 30 de vídeo puede implementarse a través de un circuito lógico que puede incluir un búfer de imágenes (por ejemplo, a través de la(s) unidad(es) 420 de procesamiento o (el)los almacenamiento(s) 44 de memoria) y una unidad de procesamiento de gráficos (por ejemplo, a través de la(s) unidad(es) 46 de procesamiento). La unidad de procesamiento de gráficos puede estar acoplada comunicativamente al búfer de imágenes. La unidad de procesamiento de gráficos puede incluir un

60

65

decodificador 30 de vídeo implementado a través del circuito 47 lógico para incorporar los diversos módulos como se analizó con respecto a la FIG. 3 y/o cualquier otro sistema o subsistema decodificador descrito en la presente memoria.

5 En algunos ejemplos, la antena 42 del sistema 40 de codificación de vídeo puede configurarse para recibir un flujo de bits codificado de datos de vídeo. Como se analizó, el flujo de bits codificado puede incluir datos, indicadores, valores de índice, datos de selección de modo o similares asociados con la codificación de un fotograma de vídeo como se analizó en la presente memoria, tal como datos asociados con la partición de codificación (por ejemplo, coeficientes de transformación o coeficientes de transformación cuantificados, indicadores opcionales (como se analizó), y/o datos que definen la partición de codificación). El sistema 40 de codificación de vídeo también puede incluir un decodificador 30 de vídeo acoplado a la antena 42 y configurado para decodificar el flujo de bits codificado. El dispositivo 45 de visualización está configurado para presentar fotogramas de vídeo.

15 Codificador y método de codificación

La FIG. 2 muestra un diagrama de bloques esquemático/conceptual de un codificador 20 de vídeo de ejemplo que está configurado para implementar las técnicas de la presente solicitud. En el ejemplo de la FIG. 2, el codificador 20 de vídeo comprende una unidad 204 de cálculo residual, una unidad 206 de procesamiento de transformación, una unidad 208 de cuantificación, una unidad 210 de cuantificación inversa y una unidad 212 de procesamiento de transformación inversa, una unidad 214 de reconstrucción, un búfer 216, una unidad 220 de filtro de bucle, un búfer de imágenes decodificadas (DPB) 230, una unidad 260 de procesamiento de predicción y una unidad 270 de codificación de entropía. La unidad 260 de procesamiento de predicción puede incluir una unidad 244 de interpredicción, una unidad 254 de intrapredicción y una unidad 262 de selección de modo. La unidad 244 de interpredicción puede incluir una unidad de estimación de movimiento y una unidad de compensación de movimiento (no mostrada). Un codificador 20 de vídeo como se muestra en la FIG. 2 también puede denominarse codificador de vídeo híbrido o codificador de vídeo según un códec de vídeo híbrido.

30 Por ejemplo, la unidad 204 de cálculo residual, la unidad 206 de procesamiento de transformación, la unidad 208 de cuantificación, la unidad 260 de procesamiento de predicción y la unidad 270 de codificación de entropía forman una ruta de señal directa del codificador 20, mientras que, por ejemplo, la unidad 210 de cuantificación inversa, la unidad 212 de procesamiento de transformación inversa, la unidad 214 de reconstrucción, el búfer 216, el filtro 220 de bucle, el búfer de imágenes decodificadas (DPB) 230, la unidad 260 de procesamiento de predicción forman una ruta de señal inversa del codificador, en donde la ruta de señal inversa del codificador corresponde a la ruta de señal del decodificador (consulte el decodificador 30 en la FIG. 3).

El codificador 20 está configurado para recibir, por ejemplo, por la entrada 202, una imagen 201 o un bloque 203 de la imagen 201, por ejemplo, una imagen de una secuencia de imágenes que forman un vídeo o secuencia de vídeo. El bloque 203 de imagen también puede denominarse bloque de imagen actual o bloque de imagen a codificar, y la imagen 201 imagen actual o imagen a codificar (en particular en la codificación de vídeo para distinguir la imagen actual de otras imágenes, por ejemplo, imágenes codificadas y/o decodificadas anteriormente de la misma secuencia de vídeo, es decir, la secuencia de vídeo que también comprende la imagen actual).

45 Particionamiento

Las realizaciones del codificador 20 pueden comprender una unidad de particionamiento (no representada en la FIG. 2) configurada para dividir la imagen 201 en una pluralidad de bloques, por ejemplo, bloques como el bloque 203, generalmente en una pluralidad de bloques que no se superponen. La unidad de particionamiento puede configurarse para usar el mismo tamaño del bloque para todas las imágenes de una secuencia de vídeo y la cuadrícula correspondiente que define el tamaño del bloque, o para cambiar el tamaño del bloque entre imágenes, subconjuntos o grupos de imágenes, y dividir cada imagen en los bloques correspondientes.

55 En un ejemplo, la unidad 260 de procesamiento de predicción del codificador 20 de vídeo puede configurarse para realizar cualquier combinación de las técnicas de particionamiento descritas anteriormente.

Al igual que la imagen 201, el bloque 203 nuevamente es o puede ser considerado como un conjunto o matriz bidimensional de muestras con valores de intensidad (valores de muestra), aunque de menor dimensión que la imagen 201. En otras palabras, el bloque 203 puede comprender, por ejemplo, una matriz de muestras (por ejemplo, una matriz de luma en el caso de una imagen 201 monocromática) o tres matrices de muestras (por ejemplo, una matriz de luma y dos matrices de croma en el caso de una imagen 201 en color) o cualquier otro número y/o tipo de matrices dependiendo del formato de color aplicado. El número de muestras en dirección horizontal y vertical (o eje) del bloque 203 define el tamaño del bloque 203.

65 El codificador 20, como se muestra en la FIG. 2, está configurado para codificar la imagen 201 bloque por

bloque, por ejemplo, la codificación y la predicción se realizan por bloque 203.

Cálculo residual

5 La unidad 204 de cálculo residual está configurada para calcular un bloque 205 residual basándose en el bloque 203 de imagen y en un bloque 265 de predicción (más adelante se proporcionan más detalles sobre el bloque 265 de predicción), por ejemplo, restando los valores de muestra del bloque 203 de imagen, muestra por muestra (píxel por píxel) para obtener el bloque 205 residual en el dominio de la muestra.

10

Transformación

15 La unidad 206 de procesamiento de transformación está configurada para aplicar una transformación, por ejemplo, una transformación de coseno discreta (DCT) o una transformación de seno discreta (DST), en los valores de muestra del bloque 205 residual para obtener coeficientes 207 de transformación en un dominio de transformación. Los coeficientes 207 de transformación también pueden denominarse coeficientes residuales de transformación y representan el bloque 205 residual en el dominio de transformación.

20 La unidad 206 de procesamiento de transformación puede configurarse para aplicar aproximaciones enteras de DCT/DST, tales como las transformaciones especificadas para HEVC/H.265. En comparación con una transformación DCT ortogonal, tales aproximaciones de enteros generalmente se escalan mediante un determinado factor. Con el fin de preservar la norma del bloque residual que se procesa mediante transformaciones directas e inversas, se aplican factores de escalado adicionales como parte del proceso de transformación. Los factores de escalado se eligen generalmente en función de ciertas restricciones, como que  
25 los factores de escalado sean una potencia de dos para la operación de desplazamiento, la profundidad de bits de los coeficientes de transformación, el equilibrio entre la precisión y los costes de implementación, etc. Los factores de escalado específicos se especifican, por ejemplo, para la transformación inversa, por ejemplo, mediante la unidad 212 de procesamiento de transformación inversa, en un decodificador 30 (y la  
30 transformación inversa correspondiente, por ejemplo, mediante la unidad 212 de procesamiento de transformación inversa en un codificador 20) y los factores de escalado correspondientes para la transformación directa, por ejemplo, mediante la unidad 206 de procesamiento de transformación, en consecuencia, se puede especificar un codificador 20.

35 Cuantificación

La unidad 208 de cuantificación está configurada para cuantificar los coeficientes 207 de transformación para obtener coeficientes 209 de transformación cuantificados, por ejemplo, aplicando cuantificación escalar o cuantificación vectorial. Los coeficientes 209 de transformación cuantificados también pueden denominarse coeficientes 209 residuales cuantificados. El proceso de cuantificación puede reducir la profundidad de bits  
40 asociada con algunos o todos los coeficientes 207 de transformación. Por ejemplo, un coeficiente de transformación de  $n$  bits puede redondearse hacia abajo a un coeficiente de transformación de  $m$  bits durante la cuantificación, donde  $n$  es mayor que  $m$ . El grado de cuantificación puede modificarse ajustando un parámetro de cuantificación (QP). Por ejemplo, para la cuantificación escalar, se pueden aplicar escalados diferentes para lograr una cuantificación más fina o más gruesa. Los tamaños de los escalones de  
45 cuantificación menores corresponden a una cuantificación más fina, mientras que los tamaños de los escalones de cuantificación más grandes corresponden a una cuantificación más gruesa. El tamaño de los escalones de cuantificación aplicables puede indicarse mediante un parámetro de cuantificación (QP). El parámetro de cuantificación puede ser, por ejemplo, un índice de un conjunto predefinido de tamaños de escalones de cuantificación aplicables. Por ejemplo, los parámetros de cuantificación pequeños pueden corresponder a una  
50 cuantificación fina (tamaños de escalones de cuantificación pequeños) y los parámetros de cuantificación grandes pueden corresponder a una cuantificación gruesa (tamaños de escalones de cuantificación grandes) o viceversa. La cuantificación puede incluir la división por un tamaño del escalón de cuantificación y la decuantificación correspondiente o inversa, por ejemplo, mediante la cuantificación 210 inversa, puede incluir la multiplicación por el tamaño del escalón de cuantificación. Las realizaciones según algunos estándares, por  
55 ejemplo, HEVC, pueden configurarse para usar un parámetro de cuantificación para determinar el tamaño del escalón de cuantificación. Generalmente, el tamaño del escalón de cuantificación puede calcularse basándose en un parámetro de cuantificación usando una aproximación de punto fijo de una ecuación que incluye la división. Pueden introducirse factores de escalado adicionales para la cuantificación y la decuantificación para restaurar la norma del bloque residual, que podría modificarse debido al escalado usado en la aproximación de punto fijo de la ecuación para el tamaño del escalón de cuantificación y el parámetro de cuantificación. En  
60 una implementación de ejemplo, se pueden combinar el escalado de la transformación inversa y la decuantificación. Alternativamente, pueden usarse tablas de cuantificación personalizadas y señalarse desde un codificador a un decodificador, por ejemplo, en un flujo de bits. La cuantificación es una operación con pérdida, en donde la pérdida aumenta al aumentar los tamaños de los escalones de cuantificación.

65

La unidad 210 de cuantificación inversa está configurada para aplicar la cuantificación inversa de la unidad 208

de cuantificación en los coeficientes cuantificados para obtener los coeficientes 211 decuantificados, por ejemplo, aplicando el inverso del esquema de cuantificación aplicado por la unidad 208 de cuantificación basándose en o usando el mismo tamaño del escalón de cuantificación que la unidad 208 de cuantificación. Los coeficientes 211 decuantificados también pueden denominarse coeficientes 211 residuales  
 5 decuantificados y corresponden - aunque generalmente no son idénticos a los coeficientes de transformación debido a la pérdida por cuantificación - a los coeficientes 207 de transformación.

La unidad 212 de procesamiento de transformación inversa está configurada para aplicar la transformación inversa de la transformación aplicada por la unidad 206 de procesamiento de transformación, por ejemplo, una  
 10 transformación de coseno discreta (DCT) inversa o una transformación de seno discreta (DST) inversa, para obtener un bloque 213 de transformación inversa en el dominio de la muestra. El bloque 213 de transformación inversa también puede denominarse bloque 213 decuantificado de transformación inversa o bloque 213 residual de transformación inversa.

La unidad 214 de reconstrucción (por ejemplo, sumador 214) está configurada para añadir el bloque 213 de transformación inversa (es decir, el bloque 213 residual reconstruido) al bloque 265 de predicción para obtener un bloque 215 reconstruido en el dominio de la muestra, por ejemplo, añadiendo los valores de muestra del  
 15 bloque 213 residual reconstruido y los valores de muestra del bloque 265 de predicción.

Opcionalmente, la unidad 216 de búfer (o de forma abreviada, "búfer" 216), por ejemplo, un búfer 216 de línea, está configurado para almacenar en búfer o almacenar el bloque 215 reconstruido y los valores de muestra respectivos, por ejemplo para intrapredicción. En realizaciones adicionales, el codificador puede configurarse para usar bloques reconstruidos sin filtrar y/o los respectivos valores de muestra almacenados en la unidad  
 20 216 de búfer para cualquier tipo de estimación y/o predicción, por ejemplo, intrapredicción.

Las realizaciones del codificador 20 pueden configurarse de manera que, por ejemplo, la unidad 216 de búfer no solo se use para almacenar los bloques 215 reconstruidos para la intrapredicción 254, sino también para la unidad 220 de filtro de bucle (no mostrada en la FIG. 2), y/o de manera que, por ejemplo, la unidad 216 de búfer y la unidad 230 de búfer de imágenes decodificadas formen un búfer. Se pueden configurar realizaciones  
 25 adicionales para usar bloques 221 filtrados y/o bloques o muestras del búfer 230 de imágenes decodificadas (ambos no mostrados en la FIG. 2) como entrada o base para la intrapredicción 254.

La unidad 220 de filtro de bucle (o de forma abreviada "filtro de bucle" 220), está configurada para filtrar el bloque 215 reconstruido para obtener un bloque 221 filtrado, por ejemplo, para suavizar las transiciones de  
 30 píxeles o de otra forma mejorar la calidad del vídeo. La unidad 220 de filtro de bucle está destinada a representar uno o más filtros de bucle, tales como un filtro de desbloqueo, un filtro de desplazamiento adaptativo de muestras (SAO) u otros filtros, por ejemplo, un filtro bilateral o un filtro de bucle adaptativo (ALF) o filtros de nitidez y de suavizado o filtros colaborativos. Aunque la unidad 220 de filtro de bucle se muestra en la FIG. 2 como un filtro de bucle, en otras configuraciones, la unidad 220 de filtro de bucle puede implementarse  
 35 como un filtro posbucle. El bloque 221 filtrado también puede denominarse bloque 221 filtrado reconstruido. El búfer 230 de imágenes decodificadas puede almacenar los bloques de codificación reconstruidos después de que la unidad 220 de filtro de bucle realice las operaciones de filtrado en los bloques de codificación reconstruidos.

Las realizaciones del codificador 20 (respectivamente, la unidad 220 de filtro de bucle) pueden configurarse para generar parámetros de filtro de bucle (tales como información de desplazamiento adaptativa de muestra), por ejemplo, directamente o codificados por entropía a través de la unidad 270 de codificación de entropía o cualquier otra unidad de codificación de entropía, para que, por ejemplo, un decodificador 30 pueda recibir y  
 40 aplicar los mismos parámetros de filtro de bucle para la decodificación.

El búfer de imágenes decodificadas (DPB) 230 puede ser una memoria de imágenes de referencia que almacena datos de imágenes de referencia para su uso en la codificación de datos de vídeo mediante el codificador 20 de vídeo. El DPB 230 puede estar formado por cualquiera de una variedad de dispositivos de memoria, tales como memoria de acceso aleatorio dinámica (DRAM), incluida la DRAM síncrona (SDRAM), la  
 45 RAM magnetorresistiva (MRAM), la RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. El DPB 230 y el búfer 216 pueden ser proporcionados por el mismo dispositivo de memoria o dispositivos de memoria separados. En algún ejemplo, el búfer de imágenes decodificadas (DPB) 230 está configurado para almacenar el bloque 221 filtrado. El búfer 230 de imágenes decodificadas puede configurarse además para almacenar otros bloques anteriormente filtrados, por ejemplo, bloques 221 anteriormente reconstruidos y filtrados, de la  
 50 misma imagen actual o de diferentes imágenes, por ejemplo, imágenes anteriormente reconstruidas, y puede proporcionar imágenes completas anteriormente reconstruidas, es decir, decodificadas (y los correspondientes bloques y muestras de referencia) y/o una imagen actual parcialmente reconstruida (y los correspondientes bloques y muestras de referencia), por ejemplo, para interpredicción. En algún ejemplo, si el bloque 215 reconstruido se reconstruye pero sin filtrado en bucle, el búfer de imágenes decodificadas (DPB) 230 se  
 55 configura para almacenar el bloque 215 reconstruido.

5 La unidad 260 de procesamiento de predicción, también denominada unidad 260 de procesamiento de predicción de bloque, está configurada para recibir u obtener el bloque 203 (bloque 203 actual de la imagen 201 actual) y datos de imagen reconstruidos, por ejemplo, muestras de referencia de la misma imagen (actual) del búfer 216 y/o datos 231 de imágenes de referencia de una o una pluralidad de imágenes anteriormente decodificadas del búfer 230 de imágenes decodificadas, y para procesar tales datos para predicción, es decir, para proporcionar un bloque 265 de predicción, que puede ser un bloque 245 interpredicho o un bloque 255 intrapredicho.

10 La unidad 262 de selección de modo puede configurarse para seleccionar un modo de predicción (por ejemplo, un modo de intrapredicción o interpredicción) y/o un bloque 245 o 255 de predicción correspondiente para ser usado como bloque 265 de predicción para el cálculo del bloque 205 residual y para la reconstrucción del bloque 215 reconstruido.

15 Las realizaciones de la unidad 262 de selección de modo pueden configurarse para seleccionar el modo de predicción (por ejemplo, de aquellos soportados por la unidad 260 de procesamiento de predicción), que proporciona la mejor coincidencia o, en otras palabras, el residual mínimo (el residual mínimo significa una mejor compresión para la transmisión o almacenamiento) o una sobrecarga de señalización mínima (la sobrecarga de señalización mínima significa una mejor compresión para la transmisión o el almacenamiento), o que considera o equilibra ambos. La unidad 262 de selección de modo puede configurarse para determinar  
20 el modo de predicción basándose en la optimización de distorsión de tasa (RDO), es decir, seleccionar el modo de predicción que proporciona una optimización de distorsión de tasa mínima o cuya distorsión de tasa asociada cumpla al menos un criterio de selección de modo de predicción.

25 A continuación, se explicará con más detalle el procesamiento de predicción (por ejemplo, la unidad 260 de procesamiento de predicción) y la selección de modo (por ejemplo, mediante la unidad 262 de selección de modo) que realiza un codificador 20 de ejemplo.

30 Como se ha descrito anteriormente, el codificador 20 está configurado para determinar o seleccionar el modo de predicción mejor u óptimo de un conjunto de modos de predicción (predeterminados). El conjunto de modos de predicción puede comprender, por ejemplo, modos de intrapredicción y/o modos de interpredicción.

35 El conjunto de modos de intrapredicción puede comprender 35 modos de intrapredicción diferentes, por ejemplo, modos no direccionales como el modo DC (o medio) y el modo planar, o modos direccionales, por ejemplo, como se define en H.265, o puede comprender 67 modos de intrapredicción diferentes, por ejemplo, modos no direccionales como el modo DC (o medio) y el modo planar, o modos direccionales, por ejemplo, como se define en H.266 en desarrollo.

40 El conjunto de (o posibles) modos de interpredicción depende de las imágenes de referencia disponibles (es decir, imágenes anteriores al menos parcialmente decodificadas, por ejemplo, almacenadas en DBP 230) y otros parámetros de interpredicción, por ejemplo, si se usa la imagen de referencia completa o solo una parte, por ejemplo, un área de ventana de búsqueda alrededor del área del bloque actual de la imagen de referencia para buscar un bloque de referencia que coincida mejor, y/o por ejemplo, si se aplica o no la interpolación de píxeles, por ejemplo, interpolación de medio/semipel y/o de cuarto de pel.

45 Además de los modos de predicción anteriores, se puede aplicar el modo de salto y/o el modo directo.

50 La unidad 260 de procesamiento de predicción puede configurarse además para dividir el bloque 203 en subbloques o particiones de bloque más pequeñas, por ejemplo, de forma iterativa usando particionamiento de árbol cuádruple (QT), particionamiento binario (BT) o particionamiento de árbol triple (TT) o cualquier combinación de los mismos, y para realizar, por ejemplo, la predicción para cada una de las particiones de bloque o subbloques, en donde la selección de modo comprende la selección de la estructura de árbol del bloque 203 particionado y los modos de predicción aplicados a cada una de las particiones de bloque o subbloques.

55 La unidad 244 de interpredicción puede incluir una unidad de estimación de movimiento (ME) (no mostrada en la FIG. 2) y una unidad de compensación de movimiento (MC) (no mostrada en la FIG. 2). La unidad de estimación de movimiento está configurada para recibir u obtener el bloque 203 de imagen (bloque 203 de imagen actual de la imagen 201 actual) y una imagen 231 decodificada, o al menos uno o una pluralidad de bloques anteriormente reconstruidos, por ejemplo, bloques reconstruidos de uno o una pluralidad de  
60 otras/diferentes imágenes 231 anteriormente decodificadas, para estimación de movimiento. Por ejemplo, una secuencia de vídeo puede comprender la imagen actual y las imágenes 231 anteriormente decodificadas, o en otras palabras, la imagen actual y las imágenes 231 anteriormente decodificadas pueden ser parte de o formar una secuencia de imágenes que forman una secuencia de vídeo.

65 El codificador 20 puede, por ejemplo, configurarse para seleccionar un bloque de referencia de una pluralidad de bloques de referencia de las mismas o diferentes imágenes de la pluralidad de otras imágenes y

proporcionar una imagen de referencia (o índice de imagen de referencia, ...) y/o un desplazamiento (desplazamiento espacial) entre la posición (coordenadas x, y) del bloque de referencia y la posición del bloque actual como parámetros de interpredicción para la unidad de estimación de movimiento (no mostrada en la FIG. 2). Este desplazamiento también se denomina vector de movimiento (MV).

5

La unidad de compensación de movimiento está configurada para obtener, por ejemplo, recibir, un parámetro de interpredicción y para realizar una interpredicción basándose en o usando el parámetro de interpredicción para obtener un bloque 245 de interpredicción. La compensación de movimiento, realizada por la unidad de compensación de movimiento (no mostrada en la FIG. 2), puede implicar obtener o generar el bloque de predicción basándose en el vector de movimiento/bloque determinado por la estimación de movimiento, posiblemente realizando interpolaciones con precisión de subpíxel. El filtrado de interpolación puede generar muestras de píxeles adicionales a partir de muestras de píxeles conocidas, aumentando así potencialmente el número de bloques de predicción candidatos que pueden usarse para codificar un bloque de imagen. Al recibir el vector de movimiento para la PU del bloque de imagen actual, la unidad 246 de compensación de movimiento puede ubicar el bloque de predicción al que apunta el vector de movimiento en una de las listas de imágenes de referencia. La unidad 246 de compensación de movimiento también puede generar elementos de sintaxis asociados con los bloques y el segmento de vídeo para su uso por el decodificador 30 de vídeo en la decodificación de los bloques de imágenes del segmento de vídeo.

10

15

20

La unidad 254 de intrapredicción está configurada para obtener, por ejemplo recibir, el bloque 203 de imagen (bloque de imagen actual) y uno o una pluralidad de bloques anteriormente reconstruidos, por ejemplo, bloques vecinos reconstruidos, de la misma imagen para intraestimación. El codificador 20 puede, por ejemplo, estar configurado para seleccionar un modo de intrapredicción de una pluralidad de modos de intrapredicción (predeterminados).

25

Las realizaciones del codificador 20 pueden configurarse para seleccionar el modo de intrapredicción basándose en un criterio de optimización, por ejemplo, residual mínimo (por ejemplo, el modo de intrapredicción que proporciona el bloque 255 de predicción más similar al bloque 203 de imagen actual) o distorsión de tasa mínima.

30

La unidad 254 de intrapredicción está configurada además para determinar basándose en el parámetro de intrapredicción, por ejemplo, el modo de intrapredicción seleccionado, el bloque 255 de intrapredicción. En cualquier caso, después de seleccionar un modo de intrapredicción para un bloque, la unidad 254 de intrapredicción también se configura para proporcionar un parámetro de intrapredicción, es decir, información indicativa del modo de intrapredicción seleccionado para el bloque, a la unidad 270 de codificación de entropía. En un ejemplo, la unidad 254 de intrapredicción puede configurarse para realizar cualquier combinación de las técnicas de intrapredicción descritas más adelante.

35

40

La unidad 270 de codificación de entropía está configurada para aplicar un algoritmo o esquema de codificación de entropía (por ejemplo, un esquema de codificación de longitud variable (VLC), un esquema de VLC adaptable al contexto (CALVC), un esquema de codificación aritmética, una codificación aritmética binaria adaptable al contexto (CABAC), una codificación aritmética binaria adaptable al contexto basada en sintaxis (SBAC), una codificación de entropía de particionamiento de intervalo de probabilidad (PIPE) u otra técnica o metodología de codificación de entropía) en los coeficientes 209 residuales cuantificados, los parámetros de intrapredicción, un parámetro de intrapredicción, y/o parámetros de filtro de bucle, individualmente o conjuntamente (o de ninguna forma) para obtener datos 21 de imagen codificados que pueden ser emitidos por la salida 272, por ejemplo, en la forma de un flujo 21 de bits codificados. El flujo 21 de bits codificados puede transmitirse al decodificador 30 de vídeo, o archivarse para su posterior transmisión o recuperación mediante el decodificador 30 de vídeo. La unidad 270 de codificación de entropía puede configurarse además para codificar por entropía los otros elementos de sintaxis para el segmento de vídeo actual que se está codificando.

50

Se pueden usar otras variaciones estructurales del codificador 20 de vídeo para codificar el flujo de vídeo. Por ejemplo, un codificador 20 no basándose en transformación puede cuantificar la señal residual directamente sin la unidad 206 de procesamiento de transformación para ciertos bloques o fotogramas. En otra implementación, un codificador 20 puede tener la unidad 208 de cuantificación y la unidad 210 de cuantificación inversa combinadas en una sola unidad.

55

60

La FIG. 3 muestra un decodificador 30 de vídeo de ejemplo que está configurado para implementar las técnicas de esta presente solicitud. El decodificador 30 de vídeo configurado para recibir datos de imagen codificados (por ejemplo, un flujo de bits codificado) 21, por ejemplo, codificados por el codificador 100, para obtener una imagen 131 decodificada. Durante el proceso de decodificación, el decodificador 30 de vídeo recibe datos de vídeo, por ejemplo, un flujo de bits de vídeo codificado que representa bloques de imágenes de un segmento de vídeo codificado y elementos de sintaxis asociados, desde el codificador 100 de vídeo.

65

En el ejemplo de la FIG. 3, el decodificador 30 comprende una unidad 304 de decodificación de entropía, una

5 unidad 310 de cuantificación inversa, una unidad 312 de procesamiento de transformación inversa, una unidad 314 de reconstrucción (por ejemplo, un sumador 314), un búfer 316, un filtro 320 de bucle, un búfer 330 de imágenes decodificadas y una unidad 360 de procesamiento de predicción. La unidad 360 de procesamiento de predicción puede incluir una unidad 344 de interpredicción, una unidad 354 de intrapredicción y una unidad 362 de selección de modo. El decodificador 30 de vídeo puede, en algunos ejemplos, realizar una etapa de decodificación generalmente recíproca a la etapa de codificación descrita con respecto al codificador 100 de vídeo de la FIG. 2.

10 La unidad 304 de decodificación de entropía está configurada para realizar la decodificación de entropía de los datos 21 de imagen codificados para obtener, por ejemplo, los coeficientes 309 cuantificados y/o parámetros de codificación decodificados (no mostrados en la FIG. 3), por ejemplo (decodificados) alguno o todos los parámetros de interpredicción, parámetro de intrapredicción, parámetros de filtro de bucle, y/u otros elementos de sintaxis. La unidad 304 de decodificación de entropía está configurada además para enviar parámetros de interpredicción, parámetro de intrapredicción y/u otros elementos de sintaxis a la unidad 360 de procesamiento de predicción. El decodificador 30 de vídeo puede recibir los elementos de sintaxis en el nivel de segmento de vídeo y/o el nivel de bloque de vídeo.

20 La unidad 310 de cuantificación inversa puede tener una función idéntica a la unidad 110 de cuantificación inversa, la unidad 312 de procesamiento de transformación inversa puede tener una función idéntica a la unidad 112 de procesamiento de transformación inversa, la unidad 314 de reconstrucción puede tener una función idéntica a la unidad 114 de reconstrucción, el búfer 316 puede tener una función idéntica al búfer 116, el filtro 320 de bucle puede tener una función idéntica al filtro 120 de bucle, y el búfer 330 de imágenes decodificadas puede tener una función idéntica al búfer 130 de imágenes decodificadas.

25 La unidad 360 de procesamiento de predicción puede comprender una unidad 344 de intrapredicción y una unidad 354 de intrapredicción, en donde la unidad 344 de intrapredicción puede parecerse a la unidad 144 de interpredicción en función, y la unidad 354 de intrapredicción puede parecerse a la unidad 154 de intrapredicción en función. La unidad 360 de procesamiento de predicción generalmente está configurada para realizar la predicción de bloque y/u obtener el bloque 365 de predicción a partir de los datos 21 codificados y recibir u obtener (explícita o implícitamente) los parámetros relacionados con la predicción y/o la información sobre el modo de predicción seleccionado, por ejemplo, de la unidad 304 de decodificación de entropía.

35 Cuando el segmento de vídeo se codifica como un segmento intracodificado (I), la unidad 354 de intrapredicción de la unidad 360 de procesamiento de predicción está configurada para generar el bloque 365 de predicción para un bloque de imagen del segmento de vídeo actual basándose en un modo de intrapredicción señalado y datos de bloques anteriormente decodificados del fotograma o imagen actual. Cuando el fotograma de vídeo se codifica como un segmento intercodificado (es decir, B o P), la unidad 344 de interpredicción (por ejemplo, la unidad de compensación de movimiento) de la unidad 360 de procesamiento de predicción está configurada para producir bloques 365 de predicción para un bloque de vídeo del segmento de vídeo actual basándose en los vectores de movimiento y otros elementos de sintaxis recibidos de la unidad 304 de decodificación de entropía. Para la interpredicción, los bloques de predicción pueden generarse a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El decodificador 30 de vídeo puede construir las listas de fotogramas de referencia, Lista 0 y Lista 1, usando técnicas de construcción predeterminadas basándose en imágenes de referencia almacenadas en DPB 330.

45 La unidad 360 de procesamiento de predicción está configurada para determinar la información de predicción para un bloque de vídeo del segmento de vídeo actual analizando los vectores de movimiento y otros elementos de sintaxis, y usa la información de predicción para producir los bloques de predicción para el bloque de vídeo actual que se está decodificando. Por ejemplo, la unidad 360 de procesamiento de predicción usa algunos de los elementos de sintaxis recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usado para codificar los bloques de vídeo del segmento de vídeo, un tipo de segmento de interpredicción (por ejemplo, segmento B, segmento P o segmento GPB), información de construcción para una o más de las listas de imágenes de referencia para el segmento, vectores de movimiento para cada bloque de vídeo intercodificado del segmento, estado de interpredicción para cada bloque de vídeo intercodificado del segmento, y otra información para decodificar los bloques de vídeo en el segmento de vídeo actual.

60 La unidad 310 de cuantificación inversa está configurada para cuantificar inversamente, es decir, decuantificar, los coeficientes de transformación cuantificados proporcionados en el flujo de bits y decodificados por la unidad 304 de decodificación de entropía. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación calculado por el codificador 100 de vídeo para cada bloque de vídeo en el segmento de vídeo para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que debe aplicarse.

65 La unidad 312 de procesamiento de transformación inversa está configurada para aplicar una transformación inversa, por ejemplo, una DCT inversa, una transformación entera inversa o un proceso de transformación inversa conceptualmente similar, a los coeficientes de transformación con el fin de producir bloques residuales en el dominio de píxeles.

La unidad 314 de reconstrucción (por ejemplo, el sumador 314) está configurada para añadir el bloque 313 de transformación inversa (es decir, el bloque 313 residual reconstruido) al bloque 365 de predicción para obtener un bloque 315 reconstruido en el dominio de la muestra, por ejemplo, sumando los valores de muestra del bloque 313 residual reconstruido y los valores de muestra del bloque 365 de predicción.

La unidad 320 de filtro de bucle (ya sea en el bucle de codificación o después del bucle de codificación) está configurada para filtrar el bloque 315 reconstruido para obtener un bloque 321 filtrado, por ejemplo, para suavizar las transiciones de píxeles o de otro modo mejorar la calidad del vídeo. En un ejemplo, la unidad 320 de filtro de bucle puede configurarse para realizar cualquier combinación de las técnicas de filtrado descritas más adelante. La unidad 320 de filtro de bucle está destinada a representar uno o más filtros de bucle, tal como un filtro de desbloqueo, un filtro de desplazamiento adaptativo de muestra (SAO) u otros filtros, por ejemplo, un filtro bilateral o un filtro de bucle adaptativo (ALF) o unos filtros de nitidez o suavizado o filtros colaborativos. Aunque la unidad 320 de filtro de bucle se muestra en la FIG. 3 como un filtro de bucle, en otras configuraciones, la unidad 320 de filtro de bucle puede implementarse como un filtro posbucle.

Los bloques 321 de vídeo decodificados en un fotograma o imagen dados se almacenan luego en el búfer 330 de imágenes decodificadas, que almacena imágenes de referencia usadas para la compensación de movimiento posterior.

El decodificador 30 está configurado para emitir la imagen 311 decodificada, por ejemplo, a través de la salida 312, para presentación o visualización a un usuario.

Se pueden usar otras variaciones del decodificador 30 de vídeo para decodificar el flujo de bits comprimido. Por ejemplo, el decodificador 30 puede producir el flujo de vídeo de salida sin la unidad 320 de filtrado de bucle. Por ejemplo, un codificador 30 no basándose en transformación puede cuantificar de forma inversa la señal residual directamente sin la unidad 312 de procesamiento de transformación inversa para ciertos bloques o fotogramas. En otra implementación, el decodificador 30 de vídeo puede tener la unidad 310 de cuantificación inversa y la unidad 312 de procesamiento de transformación inversa combinadas en una sola unidad.

La FIG. 4 es un diagrama esquemático de un dispositivo 400 de codificación de vídeo según una realización de la descripción. El dispositivo 400 de codificación de vídeo es adecuado para implementar las realizaciones descritas como se describen en la presente memoria. En una realización, el dispositivo 400 de codificación de vídeo puede ser un decodificador tal como el decodificador 30 de vídeo de la FIG. 1A o un codificador tal como el codificador 20 de vídeo de la FIG. 1A. En una realización, el dispositivo 400 de codificación de vídeo puede ser uno o más componentes del decodificador 30 de vídeo de la FIG. 1A o el codificador 20 de vídeo de la FIG. 1A como se describió anteriormente.

El dispositivo 400 de codificación de vídeo comprende puertos 410 de entrada y unidades receptoras (Rx) 420 para recibir datos; un procesador, unidad lógica o unidad de procesamiento central (CPU) 430 para procesar los datos; unidades de transmisión (Tx) 440 y puertos 450 de salida para transmitir los datos; y una memoria 460 para almacenar los datos. El dispositivo 400 de codificación de vídeo también puede comprender componentes ópticos a eléctricos (OE) y componentes eléctricos a ópticos (EO) acoplados a los puertos 410 de entrada, unidades 420 de recepción, unidades 440 de transmisión y puertos 450 de salida para la salida o entrada de señales ópticas o eléctricas.

El procesador 430 está implementado por hardware y software. El procesador 430 puede implementarse como uno o más chips de CPU, núcleos (por ejemplo, como un procesador multinúcleo), FPGA, ASIC y DSP. El procesador 430 está en comunicación con el puerto 410 de entrada, la unidad 420 de recepción, la unidad 440 de transmisión, los puertos 450 de salida y la memoria 460. El procesador 430 comprende un módulo 470 de codificación. El módulo 470 de codificación implementa las realizaciones descritas anteriormente. Por ejemplo, el módulo 470 de codificación implementa, procesa, prepara o proporciona las diversas operaciones de codificación. La inclusión del módulo 470 de codificación por lo tanto proporciona una mejora sustancial a la funcionalidad del dispositivo 400 de codificación de vídeo y efectúa una transformación del dispositivo 400 de codificación de vídeo a un estado diferente. De manera alternativa, el módulo 470 de codificación se implementa como instrucciones almacenadas en la memoria 460 y ejecutadas por el procesador 430.

La memoria 460 comprende uno o más discos, unidades de cinta y unidades de estado sólido y puede ser usada como un dispositivo de almacenamiento de datos de desbordamiento (overflow), para almacenar programas cuando tales programas se seleccionan para su ejecución y para almacenar instrucciones y datos que se leen durante la ejecución del programa. La memoria 460 puede ser volátil o no volátil y puede ser memoria de solo lectura (ROM), memoria de acceso aleatorio (RAM), memoria ternaria de contenido direccionable (TCAM), y/o memoria de acceso aleatorio estática (SRAM).

La FIG. 5 es un diagrama de bloques simplificado de un aparato 500 que se puede usar como uno o ambos del dispositivo 310 de origen y del dispositivo 320 de destino de la FIG. 1, según una realización ejemplar. El

aparato 500 puede implementar técnicas de esta presente solicitud descritas anteriormente. El aparato 500 puede tener la forma de un sistema informático que incluya múltiples dispositivos informáticos, o la forma de un solo dispositivo informático, por ejemplo, un teléfono móvil, un ordenador tableta, un ordenador portátil, un ordenador notebook, un ordenador de escritorio, y similares.

5

Un procesador 502 en el aparato 500 puede ser una unidad de procesamiento central. Alternativamente, el procesador 502 puede ser cualquier otro tipo de dispositivo, o múltiples dispositivos, capaces de manipular o procesar información existente en la actualidad o desarrollada en el futuro. Aunque las implementaciones descritas se pueden practicar con un solo procesador como se muestra, por ejemplo, el procesador 502, se pueden lograr ventajas en velocidad y eficiencia usando más de un procesador.

10

Una memoria 504 en el aparato 500 puede ser un dispositivo de memoria de solo lectura (ROM) o un dispositivo de memoria de acceso aleatorio (RAM) en una implementación. Cualquier otro tipo adecuado de dispositivo de almacenamiento se puede usar como memoria 504. La memoria 504 puede incluir código y datos 506 a los que accede el procesador 502 usando un bus 512. La memoria 504 puede incluir además un sistema operativo 508 y programas 510 de aplicación, incluyendo los programas 510 de aplicación al menos un programa que permita al procesador 502 realizar los métodos descritos aquí. Por ejemplo, los programas 510 de aplicación pueden incluir las aplicaciones 1 a N, que además incluyen una aplicación de codificación de vídeo que realiza los métodos descritos aquí. El aparato 500 también puede incluir memoria adicional en la forma de un almacenamiento 514 secundario, que puede ser, por ejemplo, una tarjeta de memoria usada con un dispositivo informático móvil. Debido a que las sesiones de comunicación de vídeo pueden contener una cantidad significativa de información, pueden almacenarse total o parcialmente en el almacenamiento 514 secundario y cargarse en la memoria 504 según sea necesario para el procesamiento.

15

20

25

El aparato 500 también puede incluir uno o más dispositivos de salida, tales como una pantalla 518. La pantalla 518 puede ser, en un ejemplo, una pantalla sensible al tacto que combina una pantalla con un elemento sensible al tacto que puede funcionar para detectar entradas táctiles. La pantalla 518 se puede acoplar al procesador 502 a través del bus 512. Se pueden proporcionar otros dispositivos de salida que permitan a un usuario programar o de otro modo usar el aparato 500 además de o como una alternativa a la pantalla 518.

30

Cuando el dispositivo de salida es o incluye una pantalla, la pantalla se puede implementar de diversas maneras, incluyendo mediante una pantalla de cristal líquido (LCD), una pantalla de tubo de rayos catódicos (CRT), una pantalla de plasma o una pantalla de diodo emisor de luz (LED), tal como una pantalla LED orgánica (OLED).

35

El aparato 500 también puede incluir o estar en comunicación con un dispositivo 520 de detección de imágenes, por ejemplo, una cámara, o cualquier otro dispositivo 520 de detección de imágenes existente ahora o desarrollado en el futuro que pueda detectar una imagen tal como la imagen de un usuario que opera el aparato 500. El dispositivo 520 de detección de imágenes se puede colocar de manera que esté dirigido hacia el usuario que opera el aparato 500. En un ejemplo, la posición y el eje óptico del dispositivo 520 de detección de imágenes pueden configurarse de manera que el campo de visión incluya un área que está directamente adyacente a la pantalla 518 y desde la cual es visible la pantalla 518.

40

El aparato 500 también puede incluir o estar en comunicación con un dispositivo 522 de detección de sonido, por ejemplo, un micrófono, o cualquier otro dispositivo de detección de sonido existente ahora o desarrollado en el futuro que pueda detectar sonidos cerca del aparato 500. El dispositivo 522 de detección de sonido se puede colocar de manera que se dirija hacia el usuario que opera el aparato 500 y se puede configurar para recibir sonidos, por ejemplo, habla u otras expresiones, realizadas por el usuario mientras opera el aparato 500.

45

Aunque la FIG. 5 representa el procesador 502 y la memoria 504 del aparato 500 integrados en una sola unidad, se pueden utilizar otras configuraciones. Las operaciones del procesador 502 pueden distribuirse entre múltiples máquinas (cada máquina tiene uno o más procesadores) que pueden acoplarse directamente o a través de una red área local u otra red. La memoria 504 se puede distribuir entre varias máquinas, tales como una memoria basada en red o una memoria en múltiples máquinas que realizan las operaciones del aparato 500. Aunque representado aquí como un solo bus, el bus 512 del aparato 500 puede estar compuesto por múltiples buses. Además, el almacenamiento 514 secundario se puede acoplar directamente a los otros componentes del aparato 500 o se puede acceder a él a través de una red y puede comprender una única unidad integrada, tal como una tarjeta de memoria, o múltiples unidades, tales como varias tarjetas de memoria. Por lo tanto, el aparato 500 puede implementarse en una amplia variedad de configuraciones.

50

55

60

En VVC, los vectores de movimiento de los bloques intercodificados se pueden señalar de dos maneras: modo de predicción avanzada de vectores de movimiento (AMVP) o modo de fusión. Con el modo AVMP, se señala una diferencia entre el vector de movimiento real y una predicción de vector de movimiento (MVP), un índice de referencia y un índice de MVP que se refiere a una lista de candidatos AMVP, donde el índice de referencia apunta a la imagen de referencia de donde se copia el bloque de referencia para la compensación de movimiento. Para el modo de fusión, se señala un índice de fusión que hace referencia a una lista de candidatos

65

de fusión y se hereda toda la información de movimiento asociada al candidato de fusión.

Tanto para la lista de candidatos AMVP como para la lista de candidatos de fusión, se derivan de bloques codificados temporal o espacialmente vecinos. Más específicamente, la lista de candidatos de fusión se construye comprobando los siguientes cuatro tipos de candidatos MVP de fusión en orden:

1. Los candidatos de fusión espacial, que pueden determinarse a partir de cinco bloques vecinos espaciales como se representa en la FIG. 6, es decir, los bloques A0 y A1 que se ubican en la esquina inferior izquierda, los bloques B0 y B1 ubicados en la esquina superior derecha y el bloque B2 ubicado en la esquina superior izquierda.

2. Candidato de fusión temporal MVP (TMVP).

3. Combinación de candidatos de fusión bipredictivos.

4. Vector de movimiento cero que fusiona candidatos.

Una vez que el número de candidatos de fusión disponibles alcanza el máximo de candidatos de fusión permitidos señalados (por ejemplo, 5 en condiciones de prueba comunes), el proceso de construcción de la lista de candidatos de fusión termina. Cabe señalar que, los candidatos de fusión máximos permitidos pueden ser diferentes en diferentes condiciones.

Del mismo modo, para la lista de candidatos AMVP, se comprueban tres tipos de candidatos MVP en orden:

1. Hasta dos candidatos espaciales MVP, donde uno de los dos se determina a partir de los bloques B<sub>0</sub>, B<sub>1</sub> y B<sub>2</sub>, como se muestra en la FIG. 6, y el otro de los dos se determina a partir de los bloques A<sub>0</sub> y A<sub>1</sub>, como se muestra en la FIG. 6.

2. Candidatos temporales MVP (TMVP).

3. Cero candidatos MVP.

El documento JVET-K0104 introduce un método de predicción de vectores de movimiento basado en el historial (HMVP), que es un documento de trabajo del equipo conjunto de expertos en vídeo (JVET) del WP 3 de SG 16 de la ITU-T y el JTC 1/SC 29/WG 11 del ISO/IEC (accesible en <http://phenix.it-sudparis.eu/jvet/>), donde un candidato HMVP se define como la información de movimiento de un bloque anteriormente codificado. Durante el proceso de codificación/decodificación se mantiene una tabla con varios candidatos HMVP. La tabla se vacía cuando se encuentra un nuevo segmento. Cuando hay un bloque intercodificado, la información de movimiento asociada se añade a la última entrada de la tabla como un nuevo candidato HMVP. El flujo de codificación general se muestra en la FIG. 7, que incluye:

Etapas 701. cargar una tabla con candidatos HMVP;

Etapas 702, decodificar un bloque con los candidatos HMVP en la tabla cargada;

Etapas 703, actualizar la tabla con la información de movimiento decodificada cuando se decodifica el bloque.

Las etapas 701-703 se pueden realizar circularmente.

Los candidatos HMVP podrían usarse en el proceso de construcción de la lista de candidatos de fusión. Todos los candidatos HMVP desde la última entrada hasta la primera entrada de la tabla se insertan después del candidato TMVP. La poda puede aplicarse a los candidatos HMVP. Una vez que el número total de candidatos de fusión disponibles alcanza el máximo de candidatos de fusión permitidos señalado, el proceso de construcción de la lista de candidatos de fusión termina.

La operación de poda significa identificar candidatos idénticos para el predictor de movimiento en una lista y eliminar uno de los candidatos idénticos de la lista.

Del mismo modo, los candidatos HMVP también podrían usarse en el proceso de construcción de la lista de candidatos AMVP. Los vectores de movimiento de los últimos K candidatos HMVP en la tabla se insertan después del candidato TMVP. En algunas maneras de implementación, solo se usan candidatos HMVP con la misma imagen de referencia que la imagen de referencia objetivo AMVP para construir la lista de candidatos AMVP. La poda puede aplicarse a los candidatos HMVP.

Con el fin de mejorar la eficiencia del procesamiento, se introduce un procesamiento denominado procesamiento paralelo de frente de onda (WPP), donde el modo WPP permite que las filas CTU se procesen

en paralelo. En el modo WPP, cada fila CTU se procesa en relación con su fila CTU anterior (inmediatamente adyacente) usando un retardo de dos CTU consecutivas. Por ejemplo, consulte la FIG. 8, un fotograma de imagen o área de imagen consiste en una pluralidad de filas CTU, cada hilo (fila) incluye 11 CTU, es decir, el hilo 1 incluye CTU0 a CTU10, el hilo 2 incluye CTU11 a CTU 21, el hilo 3 incluye CTU22 a CTU32, el hilo 4 incluye CTU33 a 43 ... Por lo tanto, en el modo WPP, cuando finaliza el proceso de codificación/decodificación de CTU1 en el hilo 1, puede comenzar el proceso de codificación/decodificación de CTU11 en el hilo 2, de mismo modo, cuando finaliza el proceso de codificación/decodificación de CTU12 en el hilo 2, puede comenzar el proceso de codificación/decodificación de CTU22 en el hilo 3, cuando finaliza el proceso de codificación/decodificación de CTU23 en el hilo 3, puede comenzar el proceso de codificación/decodificación de CTU33 en el hilo 4, cuando finaliza el proceso de codificación/decodificación de CTU34 en el hilo 4, puede comenzar el proceso de codificación/decodificación de CTU44 en el hilo 5.

Sin embargo, cuando se combina WPP con HMVP, como se indicó anteriormente, se mantiene y actualiza una lista HMVP después del procesamiento de cada bloque de codificación, por lo que se mantiene una lista HMVP que se sigue actualizando hasta la última CTU de una fila CTU, por lo tanto, el procesamiento paralelo de frente de onda no se puede realizar ya que el hilo N debe esperar a que finalice el procesamiento de la última CTU en la fila CTU anterior.

La FIG. 9 es un diagrama de flujo que ilustra un ejemplo de operación de un decodificador de vídeo, tal como el decodificador 30 de vídeo de la FIG. 3 según una realización de la presente solicitud. Uno o más elementos estructurales del decodificador 30 de vídeo, incluyendo la unidad 344 de interpredicción, pueden configurarse para realizar las técnicas de la FIG. 9. En el ejemplo de la FIG. 9, el decodificador 30 de vídeo puede realizar las siguientes etapas:

901. Al comienzo del procesamiento de una fila CTU, se realiza la construcción/inicialización de una lista HMVP para la fila CTU.

Cuando una CTU que se va a procesar es la primera CTU (CTU inicial) de una fila CTU, se construye o inicializa la lista HMVP para la fila CTU, por lo que la primera CTU de la fila CTU se puede procesar basándose en la lista HMVP para la fila CTU.

La lista HMVP para la fila CTU puede ser construida o inicializada por la unidad 344 de interpredicción de la FIG. 3 cuando el método es un método de codificación. Alternativamente, la lista HMVP para la fila CTU puede ser construida o inicializada por la unidad 244 de interpredicción de la FIG. 2 cuando el método es un método de decodificación.

En una manera de implementación, para un fotograma de imagen, cada fila CTU puede mantenerse con una lista HMVP diferente. En otra manera de implementación, para un área de imagen, cada fila CTU puede mantenerse con una lista HMVP diferente, donde el área de imagen está formada por una pluralidad de filas CTU, donde la imagen puede ser un segmento, un mosaico o un bloque de VVC.

Cuando el bloque es una región rectangular de filas CTU dentro de un mosaico particular en una imagen, un mosaico puede dividirse en múltiples bloques, cada uno de los cuales consiste en una o más filas CTU dentro del mosaico. Un mosaico que no está dividido en múltiples bloques también se denomina bloque. Sin embargo, un bloque que es un verdadero subconjunto de un mosaico no se denomina mosaico.

Cabe señalar que, mantener una lista HMVP diferente para cada fila CTU solo significa que una lista HMVP específica puede mantenerse para una fila CTU, pero los candidatos en listas HMVP diferentes pueden ser los mismos, por ejemplo, todos los candidatos en una lista HMVP son iguales que los candidatos en otra lista HMVP, cabe señalar que, los candidatos en una lista HMVP pueden no tener redundancia; o los candidatos en listas HMVP diferentes pueden tener superposición, por ejemplo, algunos de los candidatos en una lista HMVP son iguales con algunos de los candidatos en otra lista HMVP, y algunos de los candidatos en una lista HMVP no tienen idénticos en la otra lista HMVP; o los candidatos en listas HMVP diferentes puede ser totalmente diferentes, por ejemplo, ninguno de los candidatos en una lista HMVP tiene uno idéntico en otra lista HMVP. Cabe señalar que, cuando se han procesado todas las CTU de una fila CTU, la lista HMVP mantenida para la fila CTU puede liberarse, lo que puede reducir el requisito de almacenamiento.

La presente descripción proporcionó las siguientes maneras de construir/inicializar la lista HMVP:

Manera 1: al comienzo del procesamiento de una fila CTU, la lista HMVP correspondiente se vacía o se establece en valores predeterminados. Los valores predeterminados son candidatos predeterminados que son conocidos tanto por el codificador como por el decodificador.

Por ejemplo, la lista HMVP correspondiente se rellena con MV predeterminados tales como:

a) Los MV de la manera de unipredicción, donde un MV puede ser un vector de movimiento cero, y las imágenes

de referencia pueden incluir la primera imagen de referencia en la lista L0; y/o

5 b) Los MV de la manera de bipredicción, donde un MV puede ser un vector de movimiento cero, y las imágenes de referencia pueden incluir la primera imagen de referencia en la lista L0 y la primera imagen de referencia en la lista L1; y/o

10 c) Los MV de una imagen procesada anteriormente según el orden de procesamiento de la imagen. Más específicamente MV que pertenecen a una imagen anteriormente procesada y que están en la vecindad espacial del bloque actual, cuando la posición actual del bloque se superpone en la imagen anterior, y/o

d) Los MV de una lista temporal HMVP, donde cada imagen coubicada puede almacenar una lista temporal HMVP para cada fila CTU o para la imagen entera, por lo tanto la lista temporal HMVP puede ser usada para construir/inicializar la lista HMVP para la fila CTU actual.

15 Manera 2: al comienzo del procesamiento de una fila CTU actual, la lista HMVP correspondiente se construye/inicializa basándose en la lista HMVP de la segunda CTU de la fila CTU anterior, donde la fila CTU anterior es la fila CTU inmediatamente adyacente a la fila CTU actual y en la parte superior de la fila CTU actual.

20 Tomemos la FIG. 8 como ejemplo, donde cuando la fila CTU actual es la fila CTU del hilo 2, la fila CTU anterior es la fila CTU del hilo 1, y la segunda fila CTU de la fila anterior es CTU1; cuando la fila CTU actual es la fila CTU del hilo 3, la fila CTU anterior es la fila CTU del hilo 2, y la segunda fila CTU de la fila anterior es CTU12; cuando la fila CTU actual es la fila CTU del hilo 4, la fila CTU anterior es la fila CTU del hilo 3, y la segunda CTU de la fila anterior es CTU23; cuando la fila CTU actual es la fila CTU del hilo 5, la fila CTU anterior es la fila CTU del hilo 4, y la segunda CTU de la fila anterior es CTU34; cuando la fila CTU actual es la fila CTU del hilo 6, la fila CTU anterior es la fila CTU del hilo 5 y la segunda CTU de la fila anterior es CTU45.

25 Manera 3: al comienzo del procesamiento de una fila CTU actual, la lista HMVP correspondiente se construye/inicializa basándose en la lista HMVP de la primera CTU de la fila CTU anterior, donde la fila CTU anterior es la fila CTU inmediatamente adyacente a la fila CTU actual y en la parte superior de la fila CTU actual.

30 Tomemos la FIG. 8 como ejemplo, donde cuando la fila CTU actual es la fila CTU del hilo 2, la fila CTU anterior es la fila CTU del hilo 1, y la primera fila CTU de la fila anterior es CTU0; cuando la fila CTU actual es la fila CTU del hilo 3, la fila CTU anterior es la fila CTU del hilo 2, y la primera CTU de la fila anterior es CTU 11; cuando la fila CTU actual es la fila CTU del hilo 4, la fila CTU anterior es la fila CTU del hilo 3, y la primera CTU de la fila anterior es CTU22; cuando la fila CTU actual es la fila CTU del hilo 5, la fila CTU anterior es la fila CTU del hilo 4, y la primera CTU de la fila anterior es CTU33; cuando la fila CTU actual es la fila CTU del hilo 6, la fila CTU anterior es la fila CTU del hilo 5 y la primera CTU de la fila anterior es CTU44.

35 Según las maneras 1 a 3, el procesamiento de la fila CTU actual no necesita esperar el procesamiento de una fila CTU anterior de la fila CTU actual que se está terminando, por lo tanto, puede mejorar la eficiencia de procesamiento del fotograma de imagen actual.

40 902. Procesamiento de una CTU en la fila CTU basándose en la lista HMVP construida/inicializada

45 El procesamiento de la CTU puede ser un procesamiento de interpredicción que se realiza durante el proceso de decodificación, es decir, el procesamiento de la CTU puede implementarse por la unidad 344 de interpredicción de la FIG. 3. Alternativamente, el procesamiento de la CTU puede ser un procesamiento de interpredicción que se realiza durante el proceso de codificación, es decir, el procesamiento de la CTU puede implementarse por la unidad 244 de interpredicción de la FIG. 2.

50 Cabe señalar que, las maneras anteriores para construir/inicializar la lista HMVP también se pueden usar para el procesamiento HMVP normal de sin frentes de onda, por ejemplo, el procesamiento HMVP sin WPP. Como resultado, el procesamiento HMVP es idéntico independientemente de la aplicación de WPP, lo que reduce la necesidad de una implementación lógica adicional.

55 Cabe señalar que, el procesamiento de la FIG. 9 puede ser también un proceso de codificación implementado por un codificador, tal como el codificador 20 de vídeo de la FIG. 2 según una realización de la presente solicitud.

60 Además, cabe señalar que, los métodos mencionados anteriormente relativos a la combinación de frentes de onda y la predicción basándose en HMVP también pueden usarse para la intrapredicción. Es decir, se pueden usar modos intra históricos y la tabla histórica para cada fila CTU se inicializa a los valores predeterminados.

65 Por ejemplo, la inicialización de la lista HMVP para cada fila CTU en la intrapredicción se puede hacer con

modos intra predeterminados como, Planar, DC, Vertical, Horizontal, Modo 2, VDIA y DIA.

La FIG. 10 es un diagrama de flujo que ilustra una operación de ejemplo de un decodificador de vídeo o un codificador de vídeo, tal como el decodificador 30 de vídeo de la FIG. 3 según una realización de la presente solicitud y el codificador 20 de vídeo de la FIG. 2 según una realización de la presente solicitud. Uno o más elementos estructurales del decodificador 30/codificador 20 de vídeo, incluyendo la unidad 344 de interpredicción/unidad 244 de interpredicción, pueden configurarse para realizar las técnicas de la FIG. 10. En el ejemplo de la FIG. 10, el decodificador 30 de vídeo/codificador 20 de vídeo puede realizar las siguientes etapas:

Etapa 1010, inicializar una lista HMVP para una fila CTU actual cuando la CTU actual es la CTU inicial de una fila CTU actual.

Cabe señalar que, la fila CTU actual puede ser cualquier fila CTU de un fotograma de imagen que consiste en una pluralidad de filas CTU o un área de imagen (puede ser una parte de un fotograma de imagen) que consiste en una pluralidad de filas CTU. Y la fila CTU actual puede ser una cualquiera de la pluralidad de filas CTU.

Puede determinarse si la CTU actual es la CTU inicial (o la primera CTU) de la fila CTU actual basándose en el índice de la CTU actual. Por ejemplo, como se describe en la FIG. 8, cada CTU tiene un índice único, por lo que puede determinar si la CTU actual es la primera CTU de la fila CTU actual basándose en el índice de la CTU actual. Por ejemplo, las CTU con el índice de 0, 11, 22, 33, 44 o 55... son la primera CTU de las filas CTU, respectivamente. Alternativamente, tomemos la FIG. 8 como ejemplo, cada fila CTU incluye 11 CTU, es decir, el ancho de cada fila CTU es 11, por lo tanto, puede usar el ancho de la fila CTU para dividir el índice de una CTU para determinar si el resto es 0 o no, si el resto es 0, la CTU correspondiente es la primera CTU de una fila CTU; de lo contrario, si el resto no es 0, la CTU correspondiente no es la primera CTU de una fila CTU. Es decir, si el índice de una CTU % del ancho de una fila CTU = 0, la CTU es la primera CTU de la fila CTU; de lo contrario, si el índice de una CTU % del ancho de una fila CTU ≠ 0, la CTU no es la primera CTU de la fila CTU. Cabe señalar que, cuando el proceso de una fila CTU es de derecha a izquierda, si una CTU es la CTU inicial de una fila CTU puede determinarse de manera similar.

Después de inicializar la lista HMVP, la cantidad de vectores de movimiento candidatos en la lista HMVP inicializada es cero.

La inicialización se puede realizar vaciando la lista HMVP para la fila CTU actual, es decir, vaciando la lista HMVP para la fila CTU actual, en otras palabras, el número de candidatos en la lista HMVP para la fila CTU actual es cero.

En otra manera de implementación, el método puede incluir además la siguiente etapa: inicializar una lista HMVP para cada una de la pluralidad de filas CTU excepto la fila CTU actual, en donde las listas HMVP para la pluralidad de filas CTU son idénticas o diferentes.

La inicialización se puede realizar estableciendo valores predeterminados para la lista HMVP para la fila CTU actual, o inicializando la lista HMVP para la fila CTU actual basándose en una lista HMVP de una CTU de una fila CTU anterior como se describió anteriormente.

Etapa 1020, procesar la fila CTU actual basándose en la lista HMVP.

El procesamiento puede ser un proceso de interpredicción, por lo que se puede obtener un bloque de predicción. Se puede realizar una reconstrucción basándose en el bloque de predicción para obtener un bloque reconstruido, finalmente se puede obtener una imagen decodificada basándose en el bloque reconstruido. Los detalles de estos procesos se describieron anteriormente.

Como se muestra en la FIG. 8, el fotograma de imagen actual comprende una pluralidad de filas CTU, con el fin de mejorar la eficiencia de codificación/decodificación, la pluralidad de filas CTU se puede procesar en modo de procesamiento paralelo de frente de onda (WPP). Es decir, la fila CTU actual comienza a procesarse (o el procesamiento de la fila CTU actual comienza) cuando se procesa una CTU particular de una fila CTU anterior, donde la fila CTU anterior es la fila CTU inmediatamente adyacente a la fila CTU actual y en la parte superior de la fila CTU actual, donde la CTU particular de la fila CTU anterior es la segunda CTU de la fila CTU anterior; o la CTU particular de la fila CTU anterior es la primera CTU de la fila CTU anterior. Tomemos la FIG. 8 como ejemplo, cuando la fila CTU actual es el hilo 3, la fila CTU anterior es el hilo 2, la CTU particular de la fila CTU anterior puede ser CTU 12, es decir, cuando se procesa la CTU 12, el decodificador/codificador comienza a procesar la fila CTU del hilo 3, es decir, el decodificador/codificador comienza a procesar la CTU 22. Tomemos la FIG. 8 como otro ejemplo, cuando la fila CTU actual es el hilo 4, la fila CTU anterior es el hilo 3, la CTU particular de la fila CTU anterior puede ser CTU 23, es decir, cuando se procesa la CTU 23, el decodificador/codificador comienza a procesar la fila CTU del hilo 4, es decir, el decodificador/codificador comienza a procesar la CTU 33.

En una manera de implementación, el procesamiento de la fila CTU actual basándose en la lista HMVP puede incluir: procesar la CTU actual de la fila CTU actual; actualizar la lista HMVP inicializada basándose en la CTU actual procesada; y procesar la segunda CTU de la fila CTU actual basándose en la lista HMVP actualizada.

5

La FIG. 11 es un diagrama de bloques que muestra un ejemplo de un aparato 1100 de procesamiento de vídeo configurado para implementar realizaciones de la invención, el aparato 1100 de procesamiento de vídeo puede ser el codificador 20 o el decodificador 30, como se muestra en la FIG. 11, el aparato incluye:

10 Una unidad 1110 de inicialización, configurada para inicializar una lista HMVP para una fila CTU actual cuando la CTU actual es la CTU inicial (la primera CTU) de una fila CTU actual.

El detalle de la inicialización realizada por la unidad 1110 de inicialización puede hacer referencia a la etapa 1010.

15

Una unidad 1120 de procesamiento, configurada para procesar la fila CTU actual basándose en la lista HMVP.

El detalle del procesamiento realizado por la unidad 1120 de procesamiento puede hacer referencia a la etapa 1020.

20

El procesamiento puede ser un proceso de interpredicción, por lo que se puede obtener un bloque de predicción. Se puede realizar una reconstrucción basándose en el bloque de predicción para obtener un bloque reconstruido, finalmente se puede obtener una imagen decodificada basándose en el bloque reconstruido. Los detalles de estos procesos se describieron anteriormente.

25

Como se muestra en la FIG. 8, el fotograma de imagen actual comprende una pluralidad de filas CTU, con el fin de mejorar la eficiencia de codificación/decodificación, la pluralidad de filas CTU se puede procesar en modo WPP. Es decir, la fila CTU actual comienza a procesarse cuando se procesa una CTU particular de una fila CTU anterior, donde la fila CTU anterior es la fila CTU inmediatamente adyacente a la fila CTU actual y en la parte superior de la fila CTU actual, donde la CTU particular de la fila CTU anterior es la segunda CTU de la fila CTU anterior; o la CTU particular de la fila CTU anterior es la primera CTU de la fila CTU anterior. Tomemos la FIG. 8 como ejemplo, cuando la fila CTU actual es el hilo 3, la fila CTU anterior es el hilo 2, la CTU particular de la fila CTU anterior puede ser CTU 12, es decir, cuando se procesa la CTU 12, el decodificador/codificador comienza a procesar la fila CTU del hilo 3, es decir, el decodificador/codificador comienza a procesar la CTU 22. Tomemos la FIG. 8 como otro ejemplo, cuando la fila CTU actual es el hilo 4, la fila CTU anterior es el hilo 3, la CTU particular de la fila CTU anterior puede ser CTU 23, es decir, cuando se procesa la CTU 23, el decodificador/codificador comienza a procesar la fila CTU del hilo 4, es decir, el decodificador/codificador comienza a procesar la CTU 33.

30

La presente descripción describe además un codificador, que incluye circuitos de procesamiento para llevar a cabo el método de procesamiento de vídeo o el método de codificación de la presente descripción.

La presente descripción describe además un codificador, que incluye circuitos de procesamiento para llevar a cabo el método de procesamiento de vídeo o el método de codificación de la presente descripción.

35

La presente descripción describe además un producto de programa informático que comprende un código de programa para realizar el método de procesamiento de vídeo o el método de codificación de la presente descripción.

La presente descripción describe además un soporte de almacenamiento legible por ordenador que almacena instrucciones informáticas, que cuando se ejecutan por uno o más procesadores, hacen que el uno o más procesadores realicen el método de procesamiento de vídeo o el método de codificación de la presente descripción. El soporte de almacenamiento legible por ordenador es no transitorio o transitorio.

La presente descripción describe además un decodificador, que comprende uno o más procesadores; y un soporte de almacenamiento no transitorio legible por ordenador acoplado a los procesadores y que almacena programación para su ejecución por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el decodificador para llevar a cabo el método de procesamiento de vídeo o el método de codificación de la presente descripción.

40

La presente descripción describe además un codificador, que comprende uno o más procesadores; y un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para su ejecución por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el codificador para llevar a cabo el método de procesamiento de vídeo o el método de codificación de la presente descripción.

45

50

55

60

65

El proceso de inicialización de la lista HMVP se describe en la sintaxis general de datos de segmentos de VVC (Equipo Conjunto de Expertos en Vídeo (JVET) del WP 3 de SG 16 de la ITU-T y JTC 1/SC 29/WG 11 del ISO/IEC, Codificación de Vídeo Versátil (Borrador 6)), sección 7.3.8.1 de VVC reza:

slide_data() {	descriptor
for( i = 0; i < NumBricksInCurrSlice; i++ ) {	
CtbAddrInBs = FirstCtbAddrBs[ SliceBrickIdx[ i ] ]	
for(j = 0; j < NumCtusInBrick[ SliceBrickIdx[ i ] ]; j++, CtbAddrInBs++) {	
if( (j % BrickWidth[ SliceBrickIdx[ i ] ]) == 0 ) {	
NumHmvpCand = 0	
NumHmvpIbcCand = 0	
ResetIbcBuf = 1	
}	
CtbAddrInRs = CtbAddrBsToRs[ CtbAddrInBs ]	
coding_tree_unit( )	
if( entropy_coding_sync_enabled_flag && (j < NumCtusInBrick[ SliceBrickIdx[ i ] ] - 1) && ((j + 1) % BrickWidth[ SliceBrickIdx[ i ] ]) == 0 ) {	
end_of_subset_one_bit /* igual a 1 */	ae(v)
byte_alignment( )	
}	
}	
end_of_brick_one_bit /* igual a 1 */	ae(v)
if( i < NumBricksInCurrSlice - 1)	
byte_alignment( )	
}	
}	

5 Donde  $j \% \text{BrickWidth}[ \text{SliceBrickIdx}[ i ] ] = 0$  significa que la CTU con el índice j es la CTU inicial de una fila CTU, y NumHmvpCand = 0 significa que la cantidad de candidatos en la lista HMVP se establece en 0, en otras palabras, la lista HMVP se vacía.

10 El proceso de actualización de la lista HMVP se describe en la sección 8.5.2.16 de VVC (Equipo Conjunto de Expertos en Vídeo (JVET) del WP 3 de SG 16 de la ITU-T y JTC 1/SC 29/WG 11 del ISO/IEC, Codificación de Vídeo Versátil (Borrador 6)), que reza:

Las entradas a este proceso son:

15 vectores de movimiento luma con una precisión de 1/16 de muestra fraccional mvL0 y mvL1,

índices de referencia refIdxL0 y refIdxL1,

20 indicadores de utilización de listas de predicción predFlagL0 y predFlagL1,

índice de ponderación bipredicción bcwIdx.

25 El candidato MVP hMvpCand consiste en los vectores de movimiento de luma mvL0 y mvL1, los índices de referencia refIdxL0 y refIdxL1, los indicadores de utilización de la lista de predicción predFlagL0 y predFlagL1, y el índice de ponderación de bipredicción bcwIdx.

La lista de candidatos HmvpCandList se modifica usando el candidato hMvpCand siguiendo las siguientes etapas ordenadas:

30 La variable identicalCandExist se establece igual a FALSO y la variable removeldx se establece igual a 0.

Cuando NumHmvpCand es mayor que 0, para cada índice hMvpIdx con hMvpIdx = 0..NumHmvpCand - 1, se aplican las siguientes etapas hasta que identicalCandExist es igual a VERDADERO:

Cuando hMvpCand es igual a HmvpCandList[ hMvpldx ], identicalCandExist se establece igual a VERDADERO y removeldx se establece igual a hMvpldx.

La lista de candidatos HmvpCandList se actualiza de la siguiente manera:

5

Si identicalCandExist es igual a VERDADERO o NumHmvpCand es igual a 5, se aplica lo siguiente:

Para cada índice i con  $i = (\text{removeldx} + 1) \dots (\text{NumHmvpCand} - 1)$ , HmvpCandList[ i - 1 ] se establece igual a HmvpCandList[ i ].

10

HmvpCandList[ NumHmvpCand - 1 ] se establece igual a mvCand.

De lo contrario (identicalCandExist es igual a FALSO y NumHmvpCand es menor que 5), se aplica lo siguiente: HmvpCandList[ NumHmvpCand++ ] se establece igual a mvCand.

15

En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan mediante software, las funciones pueden almacenarse o transmitirse como una o más instrucciones o código en un medio legible por ordenador y ejecutarse mediante una unidad de procesamiento basado en hardware. Los medios legibles por ordenador pueden incluir soportes de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como soportes de almacenamiento de datos, o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, según un protocolo de comunicación. De esta manera, los medios legibles por ordenador generalmente pueden corresponder a (1) soportes de almacenamiento legibles por ordenador tangibles que no son transitorios o (2) un medio de comunicación tal como una señal u onda portadora. Los soportes de almacenamiento de datos pueden ser cualquier medio disponible al que puedan acceder uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta descripción. Un producto de programa informático puede incluir un medio legible por ordenador.

20

25

30

A modo de ejemplo, y no de limitación, tales soportes de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar el código de programa deseado en la forma de instrucciones o estructuras de datos y a los que pueda acceder un ordenador. Además, cualquier conexión se denomina adecuadamente medio legible por ordenador. Por ejemplo, si el software se transmite desde un sitio web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un cable de par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, DSL o tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debe entenderse que los soportes de almacenamiento legibles por ordenador y los soportes de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que se dirigen en cambio a soportes de almacenamiento tangibles no transitorios. El disco magnético y el disco, usados en la presente memoria, incluyen un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disquete y un disco blu-ray, donde los discos magnéticos generalmente reproducen datos magnéticamente, mientras que los discos reproducen datos ópticamente mediante láser. También deben incluirse combinaciones de los anteriores dentro del alcance de los medios legibles por ordenador.

35

40

45

Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señal digital (DSP), microprocesadores de propósito general, circuitos integrados de aplicación específica (ASIC), matrices lógicas programables en campo (FPGA), u otro equivalente integrado o circuitos lógicos discretos. En consecuencia, el término "procesador", tal como se usa en la presente memoria, puede referirse a cualquiera de las estructuras anteriores o cualquier otra estructura adecuada para la implementación de las técnicas descritas en la presente memoria. Además, en algunos aspectos, la funcionalidad descrita en la presente memoria puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para codificar y decodificar, o incorporados en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

50

55

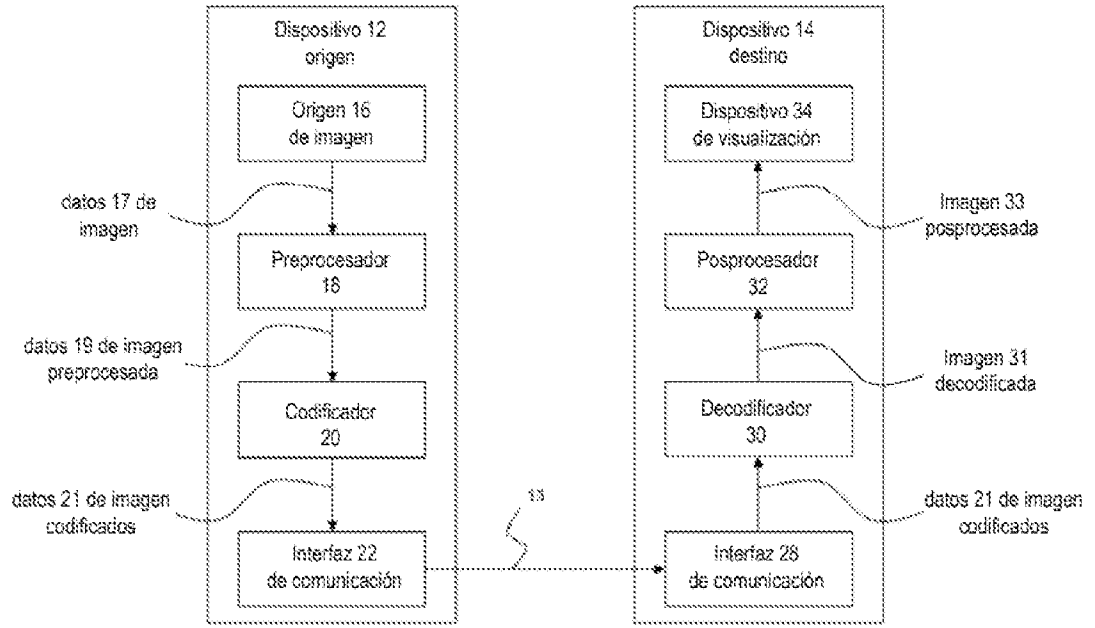
Las técnicas de esta descripción pueden implementarse en una amplia variedad de dispositivos o aparatos, que incluyen un teléfono inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). En esta descripción se describen diversos componentes, módulos o unidades para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas descritas, pero no necesariamente requieren la realización por diferentes unidades de hardware. Más bien, como se describió anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse mediante un conjunto de unidades de hardware interoperables, que incluyen uno o más procesadores como se describió anteriormente, junto con software y/o firmware adecuados.

60

65

**REIVINDICACIONES**

1. Un método de procesamiento de vídeo, caracterizado por que comprende:
- 5 al comienzo del procesamiento de cada fila de unidad de árbol de codificación, CTU, de un área de imagen que consiste en una pluralidad de filas CTU, inicializar una lista de predicción de vector de movimiento basado en el historial, HMVP, para cada una de la pluralidad de filas CTU, en donde las listas HMVP inicializadas están vacías, en donde cada fila CTU se mantiene con una lista HMVP diferente;
- 10 procesar la pluralidad de filas CTU basándose en las listas HMVP, en donde una fila CTU actual de la pluralidad de filas CTU se procesa basándose en una lista HMVP inicializada para la fila CTU actual.
2. El método según la reivindicación 1, en donde el procesamiento de una fila CTU actual de la pluralidad de filas CTU basándose en la lista HMVP comprende:
- 15 procesar una CTU actual de la fila CTU actual;
- actualizar la lista HMVP inicializada basándose en la CTU actual procesada; y
- 20 procesar la segunda CTU de la fila CTU actual basándose en la lista HMVP actualizada.
3. El método según la reivindicación 1, en donde la lista HMVP de una fila CTU actual de la pluralidad de filas CTU se actualiza según una CTU procesada de la fila CTU actual.
- 25 4. Un soporte de almacenamiento legible por ordenador que almacena instrucciones informáticas, que cuando son ejecutadas por uno o más procesadores, hacen que el uno o más procesadores realicen el método según una cualquiera de las reivindicaciones 1 a 3.
5. Un decodificador, que comprende:
- 30 uno o más procesadores; y
- un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para su ejecución por los procesadores, en donde la programación, cuando la ejecutan los procesadores, configura el decodificador para llevar a cabo el método según una cualquiera de las reivindicaciones 1 a 3.
- 35 6. Un codificador, que comprende:
- 40 uno o más procesadores; y
- un soporte de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena programación para su ejecución por los procesadores, en donde la programación, cuando la ejecutan los procesadores, configura el codificador para llevar a cabo el método según una cualquiera de las reivindicaciones 1 a 3.
- 45



10

Fig. 1A

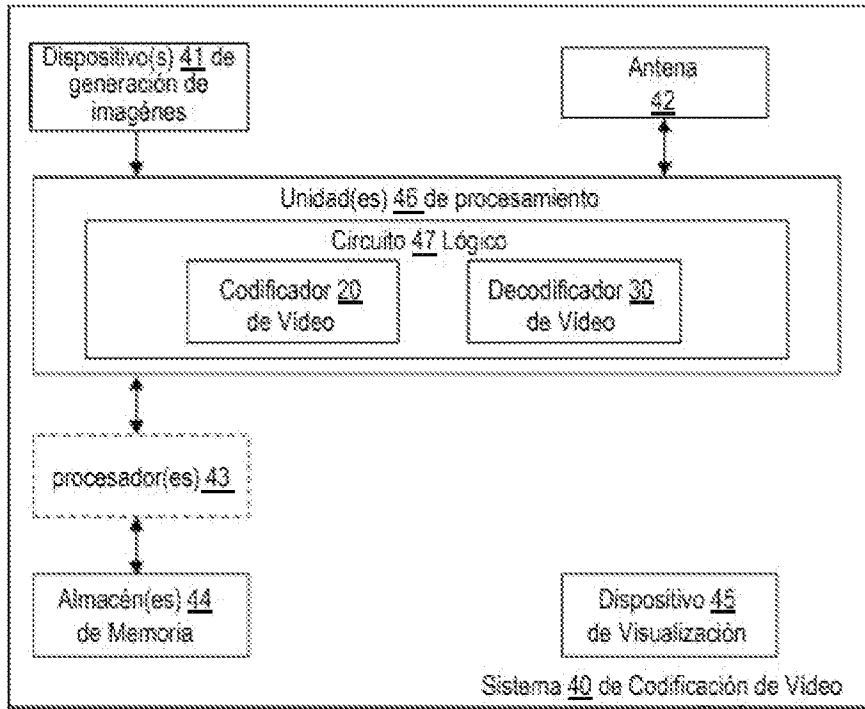


Fig. 1B

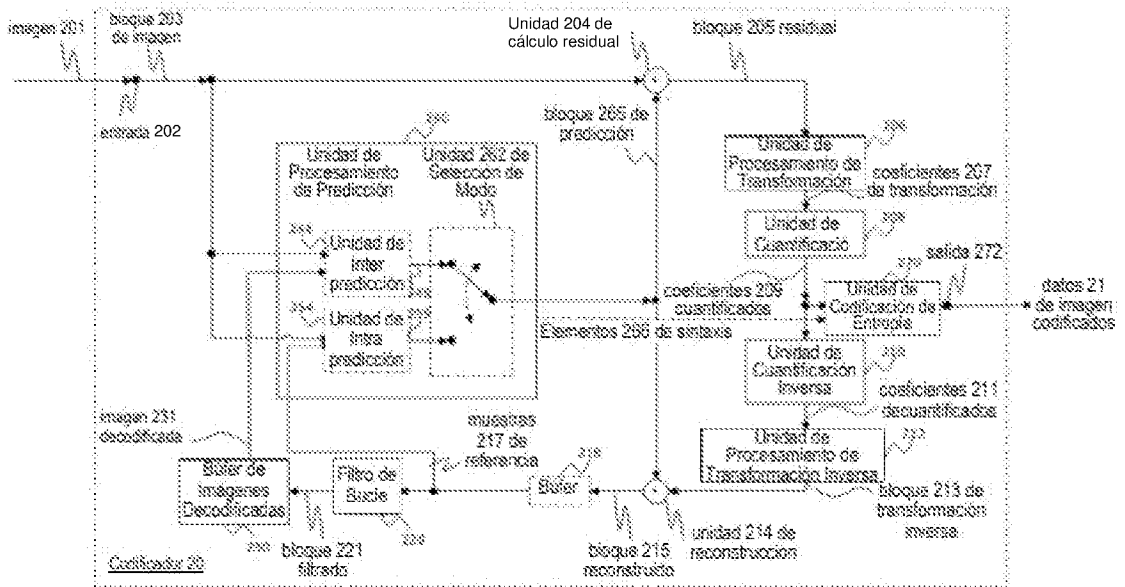


Fig. 2

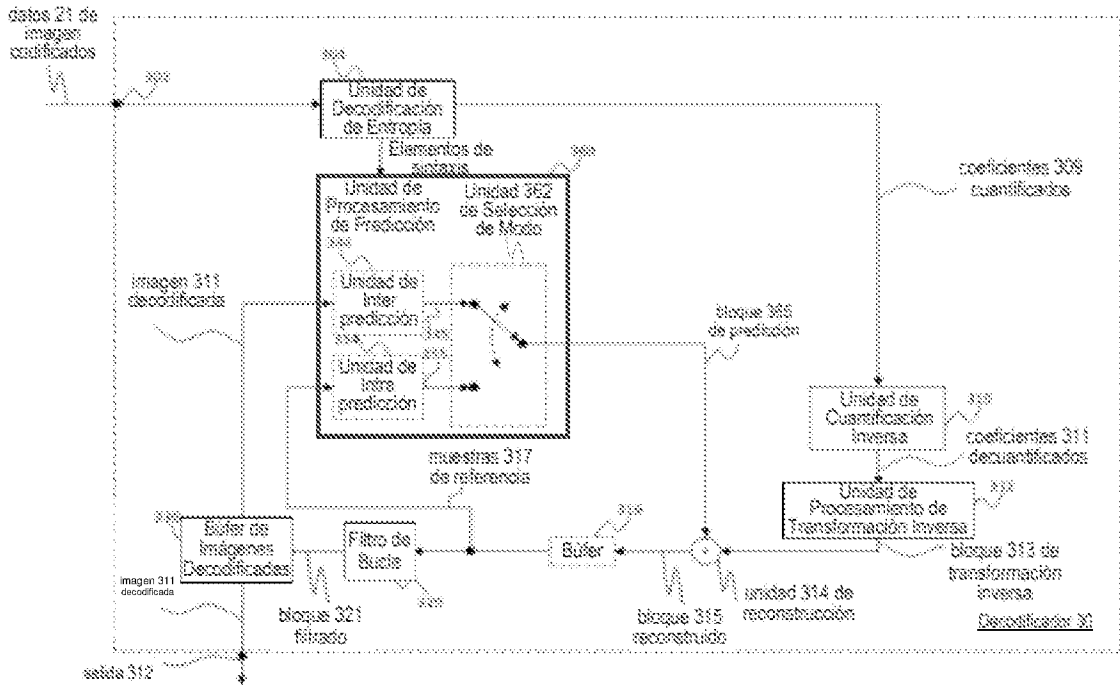


Fig. 3

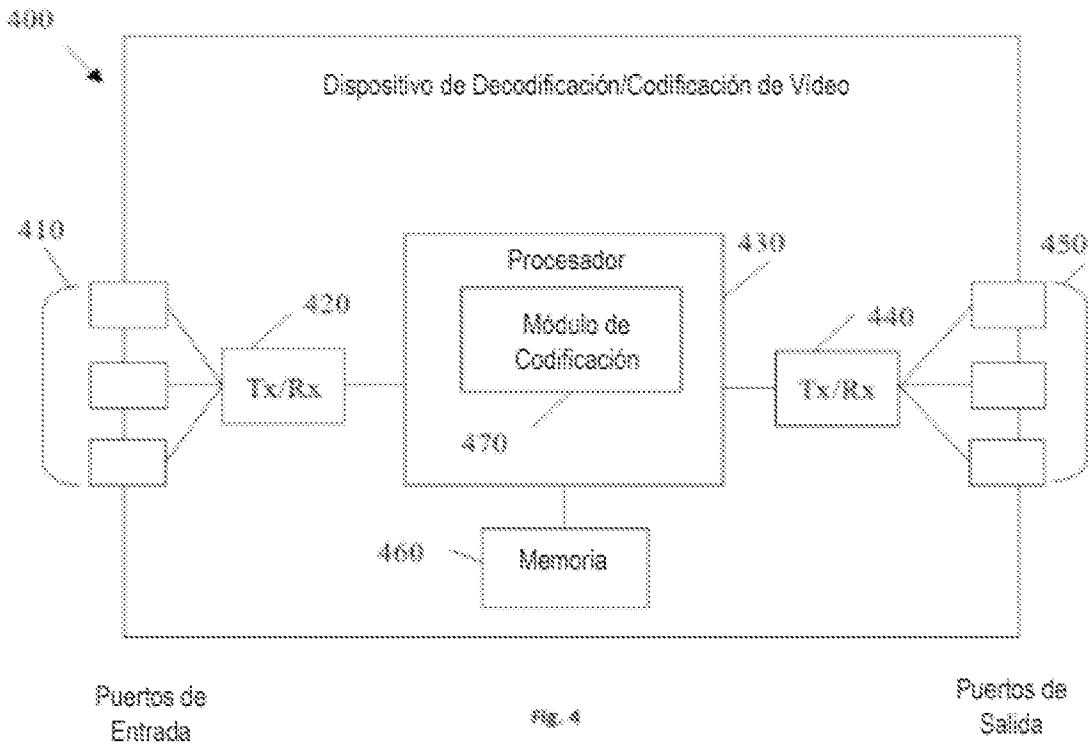


Fig. 4

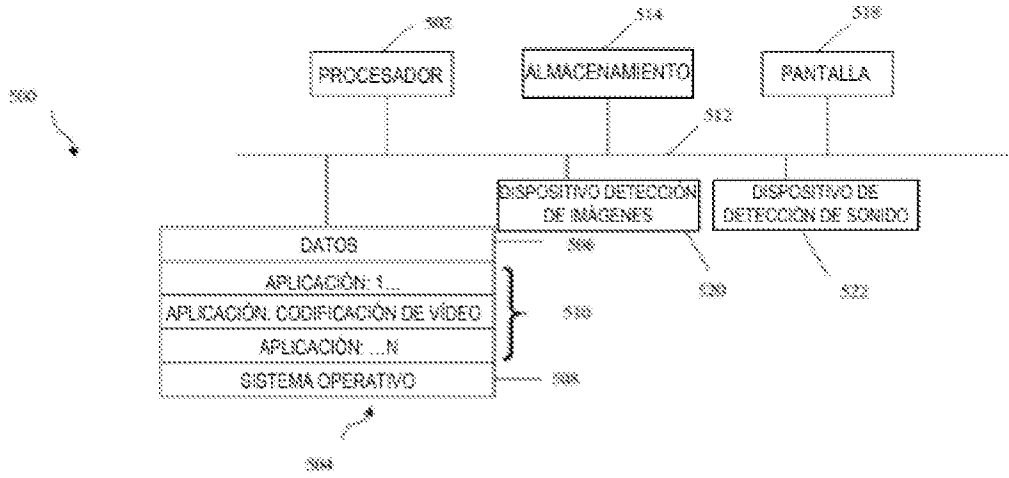


Fig. 5

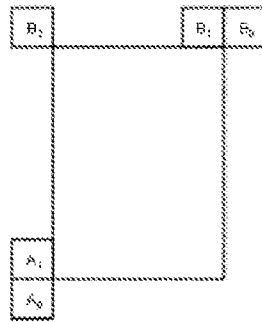


Fig. 6

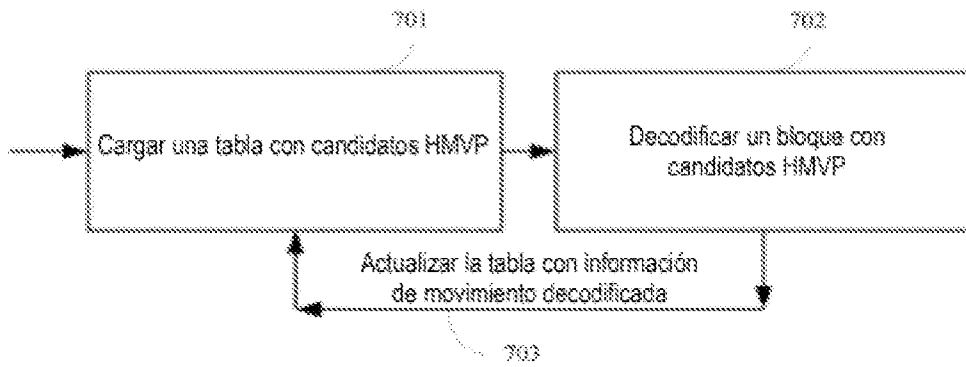


Fig. 7

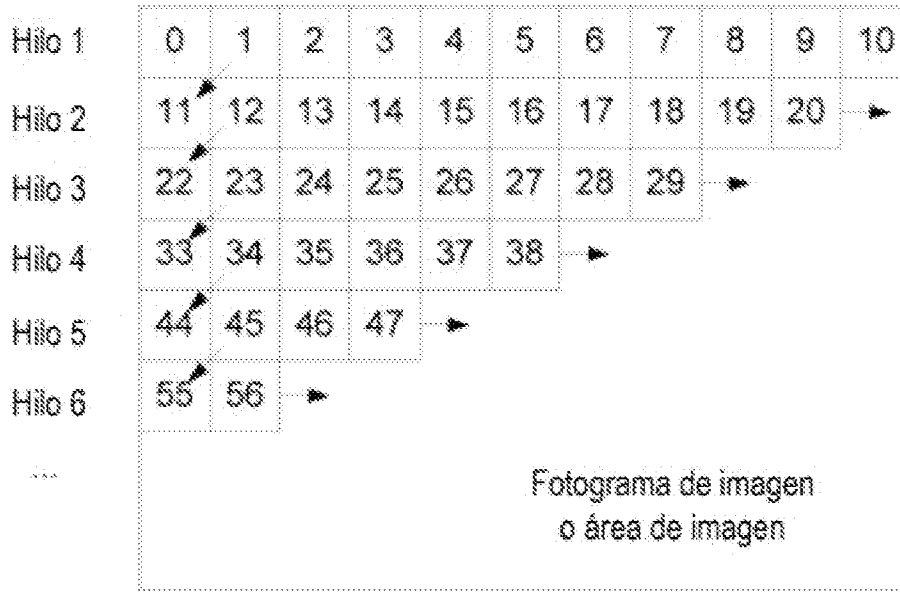


Fig. 8

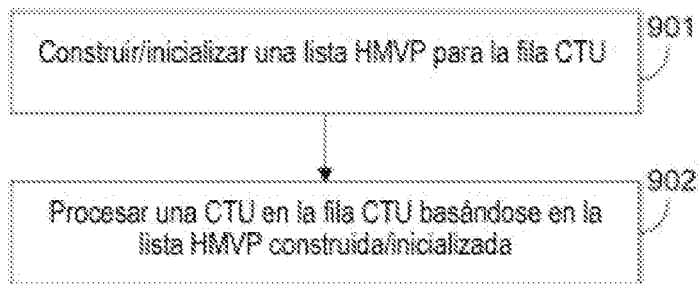


Fig. 9

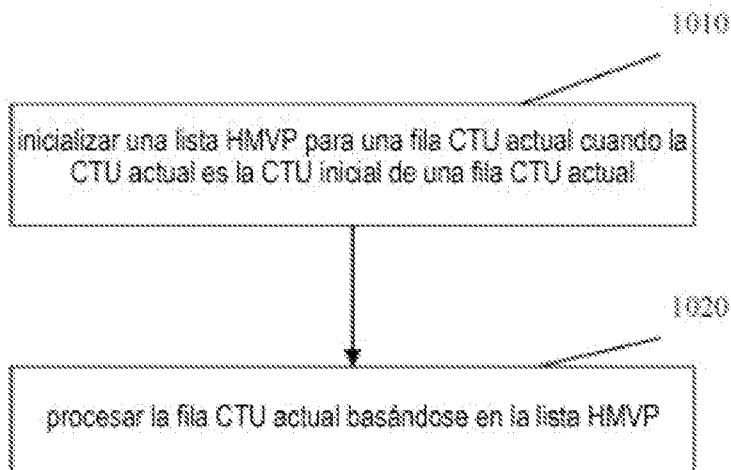


Fig. 10

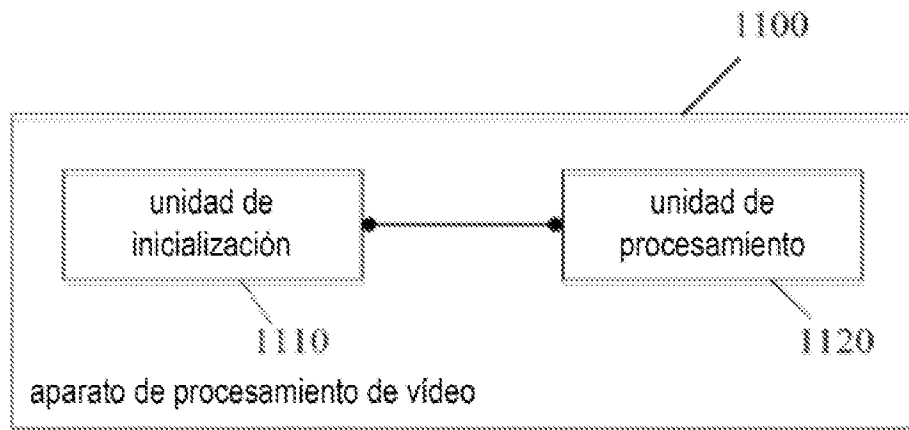


Fig. 11

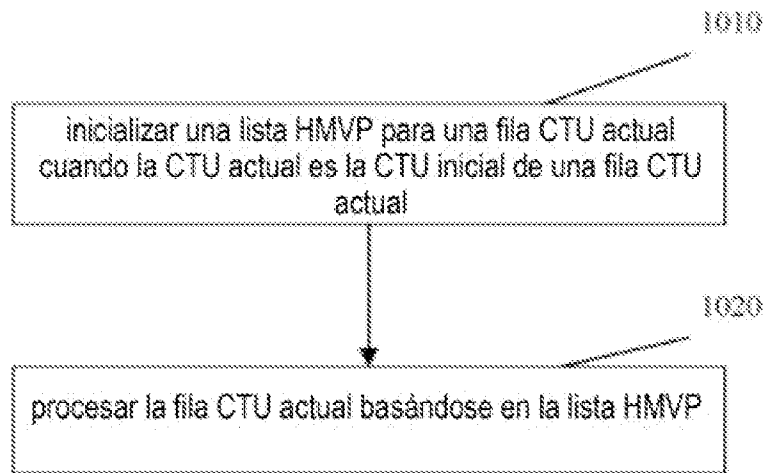


Fig. 10

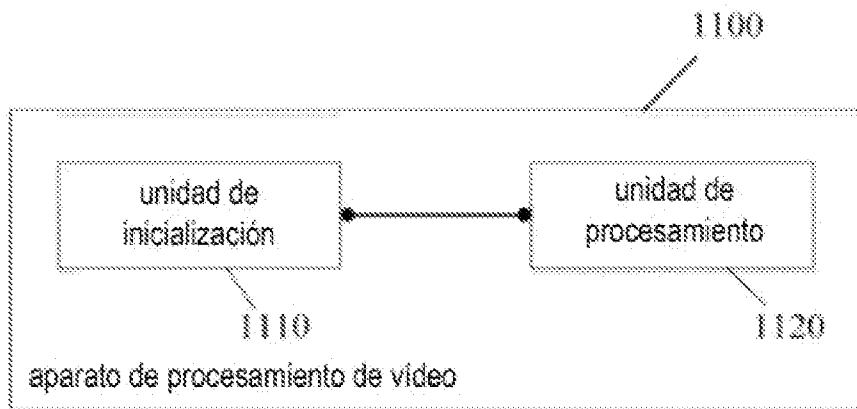


Fig. 11