

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 994 231**

51 Int. Cl.:

H04N 19/122	(2014.01)
H04N 19/147	(2014.01)
H04N 19/61	(2014.01)
H04N 19/60	(2014.01)
H04N 19/12	(2014.01)
H04N 19/17	(2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **07.10.2018** **E 22203468 (8)**

97 Fecha y número de publicación de la concesión europea: **31.07.2024** **EP 4145830**

54 Título: **Método y aparato de codificación**

30 Prioridad:

16.10.2017 US 201762572987 P
 23.02.2018 US 201862634613 P
 31.05.2018 US 201862678738 P
 27.08.2018 WO PCT/CN2018/102524

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
20.01.2025

73 Titular/es:

HUAWEI TECHNOLOGIES CO., LTD. (100.00%)
Huawei Administration Building, Bantian,
Longgang District
Shenzhen, Guangdong 518129,, CN

72 Inventor/es:

ZHAO, YIN;
YANG, HAITAO y
CHEN, JIANLE

74 Agente/Representante:

PONS ARIÑO, Ángel

ES 2 994 231 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 y aparato de codificación

- 5 Esta solicitud reivindica prioridad sobre la solicitud de patente de los Estados Unidos de América n.º 62/634,613, presentada ante la Oficina de Patentes de los Estados Unidos de América el 23 de febrero de 2018, la solicitud de patente de los Estados Unidos de América n.º 62/678,738, presentada ante la Oficina de Patentes de los Estados Unidos de América el 31 de mayo de 2018, y la solicitud de patente de los Estados Unidos de América n.º 62/572,987, presentada ante la Oficina de Patentes de los Estados Unidos de América el 16 de octubre de 2017, y la solicitud PCT/CN2018/102524, presentada ante la Oficina de Patentes de China el 27 de agosto de 2018.

Campo técnico

- 15 La presente invención se refiere al campo de las comunicaciones y, en particular, a un método y un aparato de codificación.

Antecedentes

- 20 La cantidad de datos de vídeo necesarios para representar incluso un vídeo relativamente corto puede ser sustancial, lo que puede generar dificultades cuando los datos se van a transmitir o se comunicar de otro modo a través de una red de comunicaciones con capacidad de ancho de banda limitada. Por lo tanto, los datos de vídeo generalmente se comprimen antes de comunicarse a través de las redes de telecomunicaciones modernas. El tamaño de un vídeo también podría ser un problema cuando el vídeo se almacena en un dispositivo de almacenamiento debido a que los recursos de memoria pueden ser limitados. Los dispositivos de compresión de vídeo a menudo usan *software* y/o *hardware* en la fuente para codificar los datos de vídeo antes de la transmisión o el almacenamiento, lo que reduce la cantidad de datos necesarios para representar imágenes de vídeo digital. Luego, los datos comprimidos son recibidos en el destino por un dispositivo de descompresión de vídeo que decodifica los datos de vídeo.

- 30 Con recursos de red limitados y demandas cada vez mayores de mayor calidad de vídeo, son deseables técnicas mejoradas de compresión y descompresión que mejoren la relación de compresión con poco o ningún sacrificio en la calidad de imagen.

- 35 El documento "Video coding using Variable Block-Size Spatially Varying Transforms" (de Cixun Zhang et al, abril de 2009) divulga un método de Transformaciones de Variación Espacial de Tamaño de Bloque Variable (VBSVT, por sus siglas en inglés), que selecciona y codifica un bloque de 16x4/4x16 dentro de un macrobloque de 16x16 con una transformación de 16x4/4x16 correspondiente.

- 40 El documento JCTVC-G281, "Non-CE7: Boundary Dependent Transform for Inter Predicted Residue" (de AN J ET AL, 8 de noviembre de 2011) divulga una transformación dependiente de los límites para la interpredicción, en la que se seleccionan diferentes transformaciones (DST-VII/DCT-IV o DCT-II) dependiendo de si el límite de Unidad de Transformación (TU, por sus siglas en inglés) es un límite de Unidad de Predicción (PU, por sus siglas en inglés).

- 45 El documento "Enhanced Multiple Transform for Video Coding" (de Xin Zhao ET AL, 30 de marzo de 2016) divulga un método de transformación, en el que, además de DCT-II, se introducen DST-VII, DCT-VIII, DST-I Y DCT-V para la intracodificación, y se introducen DST-VII y DCT-VIII para la intercodificación.

- 50 Compendio

Las realizaciones de la presente invención proporcionan un método y un aparato de codificación que pueden realizar una asignación de bits de cuantificación adecuada para los coeficientes espectrales de una señal de vídeo, mejorando así la calidad de una señal obtenida por un decodificador mediante la decodificación.

- 55 La invención se define en las reivindicaciones independientes. A continuación, las partes de la descripción y los dibujos que se refieren a realizaciones que no están cubiertas por las reivindicaciones no se presentan como realizaciones de la invención, sino como ejemplos útiles para comprender la invención. Para lograr el objetivo anterior, se usan las siguientes soluciones técnicas en las realizaciones de la presente invención:

- 60 Según un primer aspecto, se proporciona un método de decodificación de vídeo que incluye: obtener información de predicción de una unidad de codificación (CU, por sus siglas en inglés); cuando la CU tiene solo una unidad de transformación (TU) residual y un tamaño de la TU residual es menor que un tamaño de la CU, obtener un modo de transformación de destino de la TU residual; en donde el modo de transformación de destino especifica un modo de partición de TU de la CU, la posición de la TU residual y un tipo de transformación de la TU residual; analizar el flujo de bits recibido para obtener coeficientes de transformación

de la TU residual; aplicar una cuantificación inversa a los coeficientes de transformación de la TU residual para obtener coeficientes decuantificados; aplicar, en función del modo de transformación de destino, una transformación inversa a los coeficientes decuantificados para obtener un bloque residual de la TU residual; obtener un bloque de predicción de la CU en función de la información de predicción; obtener un bloque de vídeo en función del bloque residual y el bloque de predicción; y emitir una secuencia de vídeo, incluyendo la secuencia de vídeo un fotograma de vídeo que incluye el bloque de vídeo.

Debido a que el modo de transformación de destino puede especificar el modo de partición de TU de la CU, la posición de la TU residual y el tipo de transformación de la residual, para que no solo se pueda mejorar la eficiencia de decodificación, sino también se pueda mejorar la calidad de decodificación; y para que el modo de transformación de destino pueda especificar la información anterior, por lo que se pueden reducir los bits necesarios para transmitir la información anterior, para ahorrar el recurso de transmisión, por ejemplo, ancho de banda, para transmitir los datos de vídeo codificados.

En una primera forma de implementación posible del primer aspecto, el tamaño de la CU se indica por $W \times H$; en donde el modo de transformación de destino de la TU residual es uno de los siguientes modos:

modo de transformación Q0, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;

modo de transformación Q1, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;

modo de transformación Q2, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;

modo de transformación Q3, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;

modo de transformación QH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;

modo de transformación QH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;

modo de transformación QH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;

modo de transformación QH3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;

modo de transformación QV0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;

5 modo de transformación QV1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;

10 modo de transformación QV2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;

15 modo de transformación QV3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;

20 modo de transformación HQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es menor que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DCT-8;

25 modo de transformación HQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU inferior es menor que un tamaño de una TU superior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DST-7;

30 modo de transformación VQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8, y una transformación vertical de la TU residual es DST-7 o DCT-2;

35 modo de transformación VQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7, y una transformación vertical de la TU residual es DST-7 o DCT-2;

40 modo de transformación HH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DCT-8;

45 modo de transformación HH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DST-7;

50 modo de transformación HH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, y el tamaño de la TU superior es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-2 o una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-1;

60 modo de transformación VH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8, y una transformación vertical de la TU residual es DST-7 o DCT-2;

65 modo de transformación VH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7, y una transformación vertical de la TU residual es DST-7 o DCT-2; o

modo de transformación HH2 del modo de transformación VH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha y el tamaño de la TU izquierda es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-2 y una transformación vertical de la TU residual es DST-7 o una transformación horizontal de la TU residual es DST-1 y una transformación vertical de la TU residual es DST-7. En una forma de implementación alternativa, DST-7 puede sustituirse por DST-4, y DCT-8 puede sustituirse por DCT-4 a lo largo de toda la presente divulgación. Por consiguiente, en otra forma de implementación posible del primer aspecto, el tamaño de la CU se indica mediante $W \times H$; en donde el modo de transformación de destino de la TU residual es uno de los siguientes modos:

modo de transformación Q0, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

modo de transformación Q1, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

modo de transformación Q2, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

modo de transformación Q3, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4.

modo de transformación QH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

modo de transformación QH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño $W \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

modo de transformación QH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

modo de transformación QH3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4;

modo de transformación QV0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

modo de transformación QV1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

5 modo de transformación QV2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

10 modo de transformación QV3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4;

15 modo de transformación HQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es menor que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DCT-4;

20 modo de transformación HQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU inferior es menor que un tamaño de una TU superior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DST-4;

25 modo de transformación VQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4, y una transformación vertical de la TU residual es DST-4 o DCT-2;

30 modo de transformación VQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4, y una transformación vertical de la TU residual es DST-4 o DCT-2;

35 modo de transformación HH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DCT-4;

40 modo de transformación HH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DST-4;

45 modo de transformación HH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, y el tamaño de la TU superior es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-2 o una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-1;

50 modo de transformación VH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4, y una transformación vertical de la TU residual es DST-4 o DCT-2;

55 modo de transformación VH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4, y una transformación vertical de la TU residual es DST-4 o DCT-2; o

60 modo de transformación HH2 del modo de transformación VH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, y el tamaño de la TU izquierda es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-2 y una transformación vertical de la TU residual es DST-4 o una transformación horizontal de la TU residual es DST-1 y una transformación vertical de la TU residual es DST-4;

5 Con referencia al primer aspecto o la primera forma de implementación posible del primer aspecto, en una segunda forma de implementación posible, en donde obtener el modo de transformación de destino de la TU residual comprende: analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual; y obtener el modo de transformación de destino de la TU residual en función del índice de modo.

10 El modo de transformación de destino se puede indexar mediante el índice de modo, por lo que la eficiencia de decodificación se puede mejorar para que solo se necesite obtener un índice.

15 Con referencia a la segunda forma de implementación posible del primer aspecto, en una tercera forma de implementación posible, en donde antes de analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual, el método comprende además: determinar los modos de transformación de destino candidatos de la TU residual en función de un tamaño de la CU; en donde el análisis del flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual comprende: analizar, en función de los modos de transformación de destino candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual.

20 Determinar los modos de transformación de destino candidatos puede reducir la cantidad de modos de transformación de destino candidatos para la TU residual, para que se reduzcan los bits necesarios para transmitir el índice de modo.

25 Con referencia a la tercera forma de implementación posible del primer aspecto, en una cuarta forma de implementación posible, en donde la determinación de los modos de transformación de destino candidatos de la TU residual en función del tamaño de la CU comprende al menos uno de los siguientes:

30 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los modos de transformación de destino candidatos comprenden el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

35 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VH0 y el modo de transformación VH1;

40 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

45 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HH0 y el modo de transformación HH1;

50 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

55 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VH0 y el modo de transformación VH1;

60 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

65 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HH0 y el modo de transformación HH1;

en donde Th1, Th2 y Th3 son valores enteros predefinidos.

5 Con referencia al primer aspecto o la primera forma de implementación posible del primer aspecto, en una quinta forma de implementación posible, en donde obtener el modo de transformación de destino de la TU residual comprende: analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modo pertenece el modo de transformación de destino; analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino en el grupo de modos; y obtener el modo de transformación de destino en función del índice de grupo y el índice de modo.

10 Los modos de transformación están preagrupados, por lo que pueden reducir la cantidad de modos de transformación adecuados para una CU, no solo para mejorar la eficiencia de decodificación sino que también para ahorrar recursos de transmisión.

15 Con referencia a la quinta forma de implementación posible del primer aspecto, en una sexta forma de implementación posible, en donde el modo de transformación de destino pertenece a cualquier grupo de modos de la siguiente manera:

20 un grupo de modos comprende el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

25 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

un grupo de modos comprende el modo de transformación HH0, HH1, HH2, VH0, VH1 y VH2;

30 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

35 un grupo de modos comprende el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

40 un grupo de modos comprende el modo de transformación QV0, el modo de transformación QV1, el modo de transformación QV2, el modo de transformación QV3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

45 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

50 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

55 un grupo de modos comprende el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

60 un grupo de modos comprende el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3; o

65 un grupo de modos comprende el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

5 Con referencia a la quinta forma de implementación posible del primer aspecto, en una séptima forma de implementación posible, en donde antes de analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino, el método comprende además: determinar al menos uno de los grupos de modos candidato de la TU residual en función del tamaño de la CU; en donde el análisis del flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino comprende: analizar, en función del al menos un grupo de modos candidatos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino.

10 Determinar los modos de transformación de destino candidatos puede reducir la cantidad de modos de transformación de destino candidatos para la TU residual, para que se reduzcan los bits necesarios para transmitir el índice de modo.

15 Con referencia a la séptima forma de implementación posible del primer aspecto, en una octava forma de implementación posible, en donde la determinación de al menos uno de los grupos de modos candidatos de la TU residual en función de un tamaño de la CU comprende al menos uno de los siguientes:

20 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

25 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

30 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

35 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

40 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1;

45 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

50 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

55 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

60 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

65 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1;

60 en donde Th1, Th2 y Th3 son valores enteros predefinidos.

65 Con referencia a la cuarta forma de implementación posible o la octava forma de implementación posible del primer aspecto, en una novena forma de implementación posible, en donde Th1 es 4, 8 o 16; Th2 es 32, 64 o 128; o Th3 es 64, 128 o 256.

Con referencia al primer aspecto o cualquiera de la primera a la novena forma de implementación posible del

primer aspecto, en una décima forma de implementación posible, en donde antes de obtener el modo de transformación de destino de la TU residual, el método comprende además: determinar, en función de la información de predicción de la CU, si obtener el modo de transformación de destino de la TU residual.

5 Con referencia a la décima forma de implementación posible del primer aspecto, en una undécima forma de implementación posible, en donde la determinación, en función de la información de predicción de la CU, si obtener el modo de transformación de destino de la TU residual comprende al menos uno de los siguientes:

10 cuando un modo de predicción de la CU es un modo de intrapredicción, determinar obtener el modo de transformación de destino de la TU residual;

cuando un modo de predicción de la CU es un modo de interpredicción, determinar obtener el modo de transformación de destino de la TU residual;

15 cuando un método de predicción de la CU es un método de predicción positiva preestablecido, determinar obtener el modo de transformación de destino de la TU residual; o

cuando un método de predicción de la CU es un método de predicción negativa preestablecido, determinar no obtener el modo de transformación de destino de la TU residual.

20 Con referencia a la undécima forma de implementación posible del primer aspecto, en una duodécima forma de implementación posible, en donde el método de predicción positiva preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo de traslación; un método de predicción de combinación (*Merge*); un método avanzado de predicción de vector de movimiento con 1/4-pel de precisión de diferencia de vector de movimiento; o un método de predicción de combinación con un índice de combinación menor que 2.

30 Con referencia a la undécima o duodécima formas de implementación posibles del primer aspecto, en una decimotercera forma de implementación posible, en donde el método de predicción negativa preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo de afín; un método de predicción de combinación afín; un modo de interpredicción afín; un método avanzado de predicción de vector de movimiento con 1-pel o 4-pel de precisión de diferencia de vector de movimiento; o un método de predicción de combinación con un índice de combinación no menor que 2.

35 Con referencia al primer aspecto o uno cualquiera de la primera a la decimotercera forma de implementación posible del primer aspecto, en una decimocuarta forma de implementación posible, en donde antes de obtener el modo de transformación de destino de la TU residual, el método comprende además: determinar, en función de la información de un tamaño de la CU, si obtener el modo de transformación de destino de la TU residual.

40 Con referencia a la decimocuarta forma de implementación posible del primer aspecto, en una decimoquinta forma de implementación posible, la determinación, en función de un tamaño de la CU, si obtener el modo de transformación de destino de la TU residual comprende al menos uno de los siguientes:

45 cuando un ancho o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el modo de transformación de destino de la TU residual;

cuando un ancho o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual; o

50 cuando un ancho o una altura de una TU de la CU es mayor que un umbral de TU máximo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual.

55 Con referencia a la decimoquinta forma de implementación posible del primer aspecto, en una decimosexta forma de implementación posible, en donde el umbral de CU preestablecido es 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia; o el umbral de TU mínimo preestablecido es 4 píxeles de luminancia, 8 píxeles de luminancia o 16 píxeles de luminancia; o el umbral de TU máximo preestablecido es 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia.

60 Según un segundo aspecto, se proporciona un decodificador de vídeo, que no forma parte de la invención reivindicada, que comprende: una unidad de decodificación de entropía, configurada para analizar un flujo de bits recibido para obtener información de predicción de una unidad de codificación (CU); obtener un modo de transformación de destino de una unidad de transformación (TU) residual cuando la TU residual es la única TU residual de la CU y un tamaño de la TU residual es menor que un tamaño de la CU, en donde el modo de transformación de destino especifica un modo de partición de TU de la CU, la posición de la TU residual y un tipo de transformación de la TU residual; y analizar el flujo de bits recibido para obtener coeficientes de transformación de la TU residual; una unidad de procesamiento de cuantificación inversa, configurada para

- aplicar una cuantificación inversa a los coeficientes de transformación de la TU residual para obtener los coeficientes decuantificados; una unidad de procesamiento de transformación inversa, configurada para aplicar, en función del modo de transformación de destino, una transformación inversa a los coeficientes decuantificados para obtener un bloque residual de la TU residual; una unidad de procesamiento de predicción, configurada para obtener un bloque de predicción de la CU en función de la información de predicción; una unidad de reconstrucción, configurada para obtener un bloque de vídeo en función del bloque residual y del bloque de predicción; y una salida, configurada para emitir una secuencia de vídeo, incluyendo la secuencia de vídeo un fotograma de vídeo que incluye el bloque de vídeo.
- 5
- 10 En una primera forma de implementación posible del segundo aspecto, en donde el tamaño de la CU se indica mediante $W \times H$; en donde el modo de transformación de destino de la TU residual es uno de los siguientes modos:
- 15 modo de transformación Q0, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;
- 20 modo de transformación Q1, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;
- 25 modo de transformación Q2, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;
- 30 modo de transformación Q3, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;
- 35 modo de transformación QH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;
- 40 modo de transformación QH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;
- 45 modo de transformación QH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;
- 50 modo de transformación QH3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;
- 55 modo de transformación QV0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DCT-8;
- 60 modo de transformación QV1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior derecha, y el
- 65

tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-8;

5 modo de transformación QV2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7;

10 modo de transformación QV3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-7;

15 modo de transformación HQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es menor que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DCT-8;

20 modo de transformación HQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU inferior es menor que un tamaño de una TU superior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DST-7;

25 modo de transformación VQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2;

30 modo de transformación VQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7, y una transformación vertical de la TU residual es DST-7 o DCT-2;

35 modo de transformación HH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DCT-8;

40 modo de transformación HH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 o DCT-2, y una transformación vertical de la TU residual es DST-7;

45 modo de transformación HH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, y el tamaño de la TU superior es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DCT-2 o una transformación horizontal de la TU residual es DST-7 y una transformación vertical de la TU residual es DST-1;

50 modo de transformación VH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8, y una transformación vertical de la TU residual es DST-7 o DCT-2;

55 modo de transformación VH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-7, y una transformación vertical de la TU residual es DST-7 o DCT-2; o

60 modo de transformación HH2 del modo de transformación VH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha y el tamaño de la TU izquierda es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-

2 y una transformación vertical de la TU residual es DST-7 o una transformación horizontal de la TU residual es DST-1 y una transformación vertical de la TU residual es DST-7;

5 En una forma de implementación alternativa, DST-7 puede sustituirse por DST-4, y DCT-8 puede sustituirse por DCT-4 a lo largo de toda la presente divulgación. Por consiguiente, en otra forma de implementación posible del segundo aspecto, en donde el tamaño de la CU se indica mediante $W \times H$; en donde el modo de transformación de destino de la TU residual es uno de los siguientes modos:

10 modo de transformación Q0, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

15 modo de transformación Q1, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

20 modo de transformación Q2, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

25 modo de transformación Q3, que especifica que el modo de partición de TU de la CU es dividir la CU en 4 TU de tamaño $(W/2) \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4.

30 modo de transformación QH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

35 modo de transformación QH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte superior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la parte inferior y tiene un tamaño $W \times (H/2)$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

40 modo de transformación QH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

45 modo de transformación QH3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la parte inferior y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la posición superior y tiene un tamaño de $W \times (H/2)$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4;

50 modo de transformación QV0, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DCT-4;

55 modo de transformación QV1, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es superior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-4;

60 modo de transformación QV2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la izquierda y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3

TU está en la derecha y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4 y una transformación vertical de la TU residual es DST-4;

5 modo de transformación QV3, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde 2 TU de las 3 TU están en la derecha y tienen un tamaño de $(W/2) \times (H/2)$, y la otra TU de las 3 TU está en la izquierda y tiene un tamaño de $(W/2) \times H$, la posición de la TU residual es inferior derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-4;

10 modo de transformación HQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es menor que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DCT-4;

15 modo de transformación HQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU inferior es menor que un tamaño de una TU superior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DST-4;

20 modo de transformación VQ0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4, y una transformación vertical de la TU residual es DST-4 o DCT-2;

25 modo de transformación VQ1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4, y una transformación vertical de la TU residual es DST-4 o DCT-2;

30 modo de transformación HH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, la posición de la TU residual es superior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DCT-4;

35 modo de transformación HH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, la posición de la TU residual es inferior, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 o DCT-2, y una transformación vertical de la TU residual es DST-4;

40 modo de transformación HH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU superior es igual a un tamaño de una TU inferior, y el tamaño de la TU superior es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DCT-2 o una transformación horizontal de la TU residual es DST-4 y una transformación vertical de la TU residual es DST-1;

45 modo de transformación VH0, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-4, y una transformación vertical de la TU residual es DST-7 o DCT-2;

50 modo de transformación VH1, que especifica que el modo de partición de TU de la CU es dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformación es que una transformación horizontal de la TU residual es DST-4, y una transformación vertical de la TU residual es DST-7 o DCT-2;

55 modo de transformación HH2 del modo de transformación VH2, que especifica que el modo de partición de TU de la CU es dividir la CU en 3 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, y el tamaño de la TU izquierda es menor que un tamaño de una TU central, la posición de la TU residual es central, y el tipo de transformación es que una transformación horizontal de la TU residual es DCT-2 y una transformación vertical de la TU residual es DST-4 o una transformación horizontal de la TU residual es DST-1 y una transformación vertical de la TU residual es DST-4;

60 Con referencia al primer aspecto o la primera forma de implementación posible del segundo aspecto, en una segunda forma de implementación posible, en donde la unidad de decodificación de entropía está configurada

además para analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la residual TU; y obtener el modo de transformación de destino de la TU residual en función del índice de modo.

5 Con referencia a la segunda forma de implementación posible del segundo aspecto, en una tercera forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar modos de transformación de destino candidatos de la TU residual en función de un tamaño de la CU; y analizar, en función de los modos de transformación de destino candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual.

10 Con referencia a la tercera forma de implementación posible del segundo aspecto, en una cuarta forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar los modos de transformación de destino candidatos de la TU residual según al menos uno de los siguientes:

15 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los modos de transformación de destino candidatos comprenden el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

20 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

25 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VH0 y el modo de transformación VH1;

30 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

35 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HH0 y el modo de transformación HH1;

cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

40 cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

45 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación VH0 y el modo de transformación VH1;

50 cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

55 cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos comprenden el modo de transformación HH0 y el modo de transformación HH1;

en donde Th1, Th2 y Th3 son valores enteros predefinidos.

60 Con referencia al segundo aspecto o la primera forma de implementación posible del segundo aspecto, en una quinta forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino; y analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino en el grupo de modos; y obtener el modo de transformación de destino en función del índice de grupo y el índice de modo.

65 Con referencia a la quinta forma de implementación posible del segundo aspecto, en una sexta forma de implementación posible, en donde el modo de transformación de destino pertenece a cualquier grupo de modos

de los siguientes:

5 un grupo de modos comprende el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

10 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

un grupo de modos comprende el modo de transformación HH0, HH1, HH2, VH0, VH1 y VH2;

15 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

20 un grupo de modos comprende el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

un grupo de modos comprende el modo de transformación QV0, el modo de transformación QV1, el modo de transformación QV2, el modo de transformación QV3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

25 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

30 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

35 un grupo de modos comprende el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

un grupo de modos comprende el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

45 un grupo de modos comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3; o

un grupo de modos comprende el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

50 Con referencia a la quinta forma de implementación posible del segundo aspecto, en una séptima forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar al menos un grupo de modos candidatos de la TU residual en función de un tamaño de la CU; y analizar, en función del al menos un grupo de modos candidatos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino.

60 Con referencia a la séptima forma de implementación posible del segundo aspecto, en una octava forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar al menos un grupo de modos candidatos según al menos uno de los siguientes:

65 cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de

[Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

5 cuando un ancho de la CU está en el intervalo de [Th1×2, Th2] y una altura de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

10 cuando una altura de la CU está en el intervalo de [Th1×4, Th3] y un ancho de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

15 cuando una altura de la CU está en el intervalo de [Th1×2, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1;

20 cuando un ancho de la CU está en el intervalo de [Th1×2, Th2] y una altura de la CU está en el intervalo de [Th1×2, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

25 cuando un ancho de la CU está en el intervalo de [Th1×4, Th2] y una altura de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

30 cuando un ancho de la CU está en el intervalo de [Th1×2, Th2] y una altura de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

35 cuando una altura de la CU está en el intervalo de [Th1×4, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

40 cuando una altura de la CU está en el intervalo de [Th1×2, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1;

en donde Th1, Th2 y Th3 son valores enteros predefinidos.

45 Con referencia a la quinta u octava formas de implementación posibles del segundo aspecto, en una novena forma de implementación posible, en donde Th1 es 4, 8 o 16; Th2 es 32, 64 o 128; o Th3 es 64, 128 o 256.

50 Con referencia al segundo aspecto o una cualquiera de las formas de implementación posibles de la primera a la novena del segundo aspecto, en una décima forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar, en función de la información de predicción de la CU, si obtener el modo de transformación de destino de la TU residual.

55 Con referencia a la décima forma de implementación posible del segundo aspecto, en una undécima forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar si obtener los modos de transformación de destino de la TU residual según al menos uno de los siguientes:

cuando un modo de predicción de la CU es un modo de intrapredicción, determinar obtener el modo de transformación de destino de la TU residual;

60 cuando un modo de predicción de la CU es un modo de interpredicción, determinar obtener el modo de transformación de destino de la TU residual;

cuando un método de predicción de la CU es un método de predicción positiva preestablecido, determinar obtener el modo de transformación de destino de la TU residual; o

65 cuando un método de predicción de la CU es un método de predicción negativa preestablecido, determinar no obtener el modo de transformación de destino de la TU residual.

Con referencia a la undécima forma de implementación posible del segundo aspecto, en una duodécima forma de implementación posible, en donde el método de predicción positiva preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo de traslación; un

método de predicción de combinación; un método avanzado de predicción de vector de movimiento con 1/4-pel de precisión de diferencia de vector de movimiento; o un método de predicción de combinación con un índice de combinación menor que 2.

- 5 Con referencia a la undécima o duodécima formas de implementación posibles del segundo aspecto, en una decimotercera forma de implementación posible, en donde el método de predicción negativa preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo de afin; un método de predicción de combinación afin; un modo de interpredicción afin; un método avanzado de predicción de vector de movimiento con 1-pel o 4-pel de precisión de diferencia de vector de movimiento; o
10 un método de predicción de combinación con un índice de combinación no menor que 2.

- Con referencia al segundo aspecto o una cualquiera de las formas de implementación posibles de la primera a la decimotercera del segundo aspecto, en una decimocuarta forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar, en función de la información de la CU, si obtener el modo de transformación de destino de la TU residual.
15

- Con referencia a la decimocuarta forma de implementación posible del segundo aspecto, en una decimoquinta forma de implementación posible, en donde la unidad de decodificación de entropía está configurada además para determinar si obtener los modos de transformación de destino de la TU residual según al menos uno de los siguientes:
20

cuando un ancho o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el modo de transformación de destino de la TU residual;

- 25 cuando un ancho o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual; o

cuando un ancho o una altura de una TU de la CU es mayor que un umbral de TU máximo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual.
30

- Con referencia a la decimoquinta forma de implementación posible del segundo aspecto, en una decimosexta forma de implementación posible, en donde el umbral de CU preestablecido es 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia; o el umbral de TU mínimo preestablecido es 4 píxeles de luminancia, 8 píxeles de luminancia o 16 píxeles de luminancia; o el umbral de TU máximo preestablecido es
35 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia.

- Según un tercer aspecto, se proporciona un decodificador que comprende un circuito de procesamiento para llevar a cabo el método según una cualquiera del primer aspecto y las formas de implementación imposibles de la primera a la decimosexta del primer aspecto.
40

Según un cuarto aspecto, se proporciona un producto de programa informático que comprende un código de programa para realizar el método según uno cualquiera del primer aspecto y las formas de implementación imposibles primera a decimosexta del primer aspecto.

- 45 Según un quinto aspecto, se proporciona un decodificador que comprende: uno o más procesadores; y un medio 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 uno cualquiera del primer aspecto y las formas de implementación imposibles primera a decimosexta del primer aspecto.
50

- Realizando los métodos proporcionados por la presente divulgación, el modo de transformación de destino puede especificar el modo de partición de TU de la CU, la posición de la TU residual y el tipo de transformación de la residual, de manera que no solo se puede mejorar la eficacia de decodificación, sino también se puede mejorar la calidad de decodificación; y el modo de transformación de destino puede especificar la información anterior, por lo que se pueden reducir los bits necesarios para transmitir la información anterior, para ahorrar el recurso de transmisión, por ejemplo, ancho de banda, para transmitir los datos de vídeo codificados.
55

Breve descripción de los dibujos

- 60 Para describir las soluciones técnicas en las realizaciones de la presente invención o en la técnica anterior de manera más clara, a continuación se presentan brevemente los dibujos adjuntos requeridos para describir las realizaciones o la técnica anterior. Aparentemente, los dibujos adjuntos en la siguiente descripción muestran simplemente algunas realizaciones de la presente invención, y un experto en la técnica puede seguir obteniendo otros dibujos de estos dibujos adjuntos sin esfuerzos creativos.
65

La figura 1A es un diagrama de bloques conceptual o esquemático que ilustra un sistema de codificación de

ejemplo.

La figura 1B es un diagrama ilustrativo de otro ejemplo de sistema de codificación de vídeo.

5 La figura 2 es un diagrama de bloques esquemático/conceptual de un codificador de vídeo de ejemplo.

La figura 3 es un diagrama de bloques esquemático/conceptual de un codificador de vídeo de ejemplo.

La figura 4 es un diagrama esquemático de un dispositivo de codificación de vídeo.

10

La figura 5 es un diagrama de bloques de un aparato.

La figura 6 ilustra los modos de transformación Q0, Q1, Q2 y Q3.

15

La figura 7 ilustra los modos de transformación HQ0, HQ1, VQ0 y VQ1.

La figura 8 ilustra los modos de transformación HH0, HH1, VH0 y VH1.

La figura 9 ilustra los modos de transformación HH2 y VH2.

20

La figura 10 ilustra los modos de transformación QH0, QH1, QH2 y QH3.

La figura 11 ilustra los modos de transformación QV0, QV1, QV2 y QV3.

25

La figura 12 es un diagrama esquemático de un método de decodificación de vídeo.

La figura 13 ilustra un ejemplo de intrapredicción en la codificación de vídeo.

La figura 14 es un diagrama esquemático de un mecanismo de codificación de vídeo de ejemplo.

30

Las figuras 15-16 ilustran transformaciones de transformación de variación espacial (SVT) de ejemplo.

La figura 17 ilustra posiciones candidatas de transformación SVT de ejemplo con respecto a un bloque residual.

35

La figura 18 ilustra posiciones de transformación SVT de ejemplo con respecto a un bloque residual.

La figura 19 ilustra un ejemplo de volteo horizontal de muestra residual.

La figura 20 es un diagrama de flujo de un método de ejemplo de decodificación de vídeo con SVT dependiente de la posición.

40

La figura 21 es una ilustración de las posiciones candidatas de los bloques SVT-V y SVT-H ($M1=M2=8$).

La figura 22 es una ilustración de SVT-V y SVT-H con 3 posiciones candidatas.

45

La figura 23 es una ilustración de SVT-H y SVT-V.

La figura 24 es una ilustración de las posiciones candidatas de los bloques SVT-V y SVT-H ($M1=M2=8$).

50

La figura 25 es una ilustración de un primer grupo de coeficientes de transformación y un segundo grupo de coeficientes de transformación.

Descripción de realizaciones

55

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 mediante las reivindicaciones adjuntas.

60

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 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

65

pluralidad de unidades, cada una de las cuales realiza uno 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.

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 "trama" o "imagen" se puede 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 "descodificación" de secuencias de vídeo. La combinación de la parte de codificación y la parte de decodificación también se conoce como CODEC (codificación y decodificación).

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, mediante 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.

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 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 subsiguientes.

Como se usa en la presente memoria, el término "bloque" puede ser una parte de una imagen o un trama. 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 del UIT-T (VCEG) y del grupo de expertos en imágenes en movimiento del ISO/IEC (MPEG). 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 en función del 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 marco de partición 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 mediante una estructura de árbol cuádruple. Los nodos de hoja de árbol cuádruple se dividen además mediante 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 ninguna otra partición. 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.

- 5 En las siguientes implementaciones de un codificador 20, se describen un decodificador 30 y un sistema 10 de codificación basándose en las figuras 1 a 3.

10 La figura 1A es un diagrama de bloques conceptual o esquemático que ilustra un sistema 10 de codificación ejemplar, por ejemplo, un sistema 10 de codificación de vídeo que puede utilizar técnicas de esta presente solicitud (presente divulgación). 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 figura 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 15 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 que se ha de 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, por sus siglas en inglés)) y/o cualquier combinación de las mismas (por ejemplo, una imagen de realidad aumentada (AR, por sus siglas en inglés)).

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. 35 En espacio de color o formato RBG, 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, chroma) representan los componentes de cromaticidad o información de color. En consecuencia, una imagen en formato YCbCr comprende una matriz de muestras de luminancia de valores (Y) de muestras de luminancia y dos matrices de muestras de crominancia de valores (Cb y Cr) de crominancia. Las imágenes en formato RGB se pueden convertir o 45 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 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 comprenda o almacene una imagen previamente 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, 55 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.

60 A distinción 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.

65 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.

5

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 figura 2 o la figura 4).

10

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 decodificar o almacenar.

15

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.

20

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.

25

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.

30

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.

35

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

40

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 figura 1A que apunta del dispositivo 12 de origen al dispositivo 14 de destino, o interfaces de comunicación bidireccionales, y pueden configurarse, por ejemplo, para enviar y recibir mensajes, por ejemplo, para establecer una conexión, para acuse de recibo e intercambio de cualquier otra información relacionada con el enlace de comunicación y/o transmisión de datos, por ejemplo, transmisión de datos de imagen codificados.

45

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 figura 3 o la figura 5).

50

El posprocesador 32 del dispositivo 14 de destino está configurado para posprocesar los datos 31 de imagen decodificados (también llamados datos de imagen reconstruidos), por ejemplo, la imagen 31 decodificada, para obtener datos 33 de imagen posprocesados, por ejemplo, un 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.

55

El dispositivo 34 de visualización del dispositivo 14 de destino está configurado para recibir los datos 33 de imagen posprocesados para mostrar 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 micro LED, cristal líquido sobre silicio (LCoS), procesador de luz digital (DLP) o cualquier otro tipo de pantalla.

65

Aunque la figura 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.

Como será evidente para el experto en la técnica basándose en la descripción, la existencia y la 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 figura 1A, pueden variar dependiendo del dispositivo y la aplicación reales.

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 diversos circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP, por sus siglas en inglés), circuitos integrados de aplicación específica (ASIC, por sus siglas en inglés), agrupaciones de puertas programables de campo (FPGA, por sus siglas en inglés), 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 medio 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 (CODEC) combinado en un dispositivo respectivo.

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.

El dispositivo 12 de origen y el dispositivo 14 de destino pueden comprender cualquiera de una amplia gama de dispositivos, incluidos cualquier tipo de dispositivo portátil o fijo, por ejemplo, ordenadores de bolsillo o portátiles, teléfonos móviles, teléfonos inteligentes, tabletas u ordenadores tipo tableta, cámaras, ordenadores de escritorio, módulos de conexión, 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 pueden usar cualquier tipo de sistema operativo o no usar ninguno.

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.

En algunos casos, el sistema 10 de codificación de vídeo ilustrado en la figura 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.

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 dicho 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.

La figura 1B es un diagrama ilustrativo de otro ejemplo de sistema 40 de codificación de vídeo que incluye el codificador 20 de la figura 2 y/o el decodificador 30 de la figura 3. El sistema 40 puede implementar técnicas de acuerdo con 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.

5 Como se ilustra, el o los dispositivos 41 de generación de imágenes, la antena 42, la o las unidades 46 de procesamiento, el circuito 47 lógico, el codificador 20 de vídeo, el decodificador 30 de vídeo, el o los procesadores 43, el o los almacenamientos 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.

10 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 de forma similar 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) almacenamientos 44 de memoria pueden ser cualquier tipo de memoria, tales como memoria volátil (por ejemplo, memoria estática de acceso aleatorio (SRAM), memoria dinámica de acceso aleatorio (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 a el(los) almacenamientos 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 imagen o similar.

30 En algunos ejemplos, el codificador 100 de vídeo implementado a través de un circuito lógico puede incluir un búfer de imagen (por ejemplo, a través de, bien una o varias unidades 46 de procesamiento, bien uno o varios almacenamientos 44 de memoria) y una unidad de procesamiento de gráficos (por ejemplo, a través de una o varias unidades 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 figura 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 describen en la presente memoria.

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

50 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 trama 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 tramas de vídeo.

60 Codificador y método de codificación

65 La figura 2 muestra un diagrama de bloques esquemático/conceptual de un codificador 20 de vídeo ejemplar que está configurado para implementar las técnicas de la presente solicitud. En el ejemplo de la figura 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 figura 2 también puede denominarse codificador de vídeo híbrido o un codificador de vídeo según un códec de vídeo híbrido.

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 hacia delante 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 230 de imágenes decodificadas (DPB, por sus siglas en inglés), la unidad 260 de procesamiento de predicción forman una ruta de señal hacia atrás del codificador, en donde la ruta de señal hacia atrás del codificador corresponde a la ruta de señal del decodificador (véase el decodificador 30 en la figura 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 como 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 previamente de la misma secuencia de vídeo, es decir, la secuencia de vídeo que también comprende la imagen actual).

Particionamiento

Las realizaciones del codificador 20 pueden comprender una unidad de partición (no representada en la figura 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 partición 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.

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 partición 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 luminancia en el caso de una imagen 201 monocromática) o tres matrices de muestras (por ejemplo, una matriz de luminancia y dos matrices de crominancia 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.

El codificador 20, como se muestra en la figura 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

La unidad 204 de cálculo residual está configurada para calcular un bloque 205 residual en función del bloque 203 de imágenes y de 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 265 de predicción de los valores de muestra del bloque 203 de imágenes, muestra por muestra (píxel por píxel) para obtener el bloque 205 residual en el dominio de la muestra.

Transformación

La unidad 206 de procesamiento de transformación está configurada para aplicar una transformación, por ejemplo, una transformación de coseno discreta (DCT, por sus siglas en inglés) o una transformación de seno discreta (DST, por sus siglas en inglés), 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.

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. Para preservar la norma del bloque residual que se procesa mediante transformaciones directas e inversas, se aplican factores de escala adicionales como parte del proceso de transformación. Los factores de escala se eligen generalmente en función de ciertas restricciones, como que los factores de escala 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 escala 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 transformación inversa correspondiente, por ejemplo, mediante la unidad 212 de procesamiento de transformación inversa en un codificador 20) y los factores de escala correspondientes para la transformación directa, por ejemplo, mediante la unidad 206 de procesamiento de transformación, en consecuencia, se puede especificar un codificador 20.

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 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 escalas diferentes para lograr una cuantificación más fina o más gruesa. Los tamaños de los escalones de 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 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 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 en función de un parámetro de cuantificación usando una aproximación de punto fijo de una ecuación que incluye la división. Pueden introducirse factores de escala adicionales para la cuantificación y la decuantificación para restaurar la norma del bloque residual, que podría modificarse debido a la escala usada en la aproximación de punto fijo de la ecuación para el tamaño del escalón de cuantificación y el parámetro de cuantificación. En 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 la que la pérdida aumenta al aumentar los tamaños de los escalones de cuantificación.

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 la inversa del esquema de cuantificación aplicado por la unidad 208 de cuantificación en función de 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 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 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, sumando los valores de muestra del 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 configurada 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 216 de búfer para cualquier tipo de estimación y/o predicción, por ejemplo, intrapredicción.

5

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 figura 2), y/o de manera que, por ejemplo, la unidad 216 de búfer y la unidad 230 de búfer de imágenes decodificadas formen un solo búfer. Se pueden configurar realizaciones adicionales para usar bloques 221 filtrados y/o bloques o muestras del búfer 230 de imágenes decodificadas (ambos no mostrados en la figura 2) como entrada o base para la intrapredicción 254.

10

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 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 figura 2 como un filtro en bucle, en otras configuraciones, la unidad 220 de filtro de bucle puede implementarse como un filtro posterior al bucle. 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.

15

20

25

Las realizaciones del codificador 20 (respectivamente, la unidad 220 de filtro de bucle) pueden configurarse para emitir parámetros de filtro de bucle (tales como información de desplazamiento adaptativo de muestra), por ejemplo, directamente o codificados en 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 aplicar los mismos parámetros de filtro de bucle para la decodificación.

30

El búfer 230 de imágenes decodificadas (DPB) puede ser una memoria de imágenes de referencia que almacene 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 dinámica de acceso aleatorio (DRAM), incluida la DRAM síncrona (SDRAM), la 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 previamente filtrados, por ejemplo, bloques 221 previamente reconstruidos y filtrados, de la misma imagen actual o de diferentes imágenes, por ejemplo, imágenes previamente reconstruidas, y puede proporcionar imágenes completas previamente 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 configura para almacenar el bloque 215 reconstruido.

35

40

45

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 previamente 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.

50

55

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

60

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 proporcione 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 considere o equilibre ambos. La unidad 262 de selección de modo puede configurarse para determinar el modo de predicción en función de la optimización de distorsión de tasa (RDO), es decir, seleccionar el modo

65

de predicción que proporciona una optimización de distorsión de tasa mínima o cuya distorsión de tasa asociada cumple al menos un criterio de selección de modo de predicción.

5 A continuación, se explicarán 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 ejemplar.

10 Como se ha descrito anteriormente, el codificador 20 está configurado para determinar o seleccionar el mejor modo de predicción o uno 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.

15 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.

20 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 la interpolación de píxeles, por ejemplo, interpolación de medio/semipel y/o de cuarto de pel, o no.

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

30 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 partición de árbol cuádruple (QT), partición binaria (BT) o partición de árbol triple (TT) o cualquier combinación de las mismas, 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.

35 La unidad 244 de interpredicción puede incluir una unidad de estimación de movimiento (ME, por sus siglas en inglés) (no mostrada en la figura 2) y una unidad de compensación de movimiento (MC, por sus siglas en inglés) (no mostrada en la figura 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 previamente reconstruidos, por ejemplo, bloques reconstruidos de uno o una pluralidad de otras/diferentes imágenes 231 previamente decodificadas, para 40 estimación de movimiento. Por ejemplo, una secuencia de vídeo puede comprender la imagen actual y las imágenes 231 previamente decodificadas, o en otras palabras, la imagen actual y las imágenes 231 previamente decodificadas pueden ser parte o formar una secuencia de imágenes que forman una secuencia de vídeo.

45 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 a la unidad de estimación de movimiento (no mostrada en la figura 50 2). Este desplazamiento también se denomina vector de movimiento (MV).

55 La unidad de compensación de movimiento está configurada para obtener, por ejemplo, recibir, un parámetro de interpredicción y para realizar una interpredicción en función de 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 figura 2), puede implicar obtener o generar el bloque de predicción en función del vector de movimiento/bloque determinado por la estimación de movimiento, posiblemente realizando interpolaciones con precisión de subpíxel. El filtrado de interpolación puede generar 60 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 65 decodificación de los bloques de imágenes del segmento de vídeo.

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 previamente 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 (determinados).

5

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

10

La unidad 254 de intrapredicción está configurada además para determinar en función del 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.

15

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, por sus siglas en inglés), un esquema de VLC adaptable al contexto (CALVC, por sus siglas en inglés), un esquema de codificación aritmética binaria adaptable al contexto (CABAC, por sus siglas en inglés), una codificación aritmética binaria adaptable al contexto basada en sintaxis (SBAC, por sus siglas en inglés), una codificación de entropía de partición de intervalo de probabilidad (PIPE, por sus siglas en inglés) u otra técnica o metodología de codificación de entropía) en los coeficientes 209 residuales cuantificados, los parámetros de interpredicción, 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 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 en entropía los otros elementos de sintaxis para el segmento de vídeo actual que se está codificando.

20

25

30

Se pueden usar otras variaciones estructurales del codificador 20 de vídeo para codificar el flujo de vídeo. Por ejemplo, un codificador 20 no basado en transformación puede cuantificar la señal residual directamente sin la unidad 206 de procesamiento de transformación para ciertos bloques o tramas. 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.

35

La figura 3 muestra un decodificador 30 de vídeo ejemplar 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.

40

45

En el ejemplo de la figura 3, el decodificador 30 comprende una unidad 304 de decodificación de entropía, una 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 figura 2.

50

La unidad 304 de decodificación de entropía está configurada para realizar la decodificación de entropía en los datos 21 de imagen codificados para obtener, por ejemplo, los coeficientes 309 cuantificados y/o parámetros de codificación decodificados (no mostrados en la figura 3), por ejemplo, (decodificados) cualquiera de 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.

55

60

La unidad 310 de cuantificación inversa puede tener una función idéntica a la de la unidad 110 de cuantificación inversa, la unidad 312 de procesamiento de transformación inversa puede tener una función idéntica a la de la unidad 112 de procesamiento de transformación inversa, la unidad 314 de reconstrucción puede tener una

65

función idéntica a la de la unidad 114 de reconstrucción, el búfer 316 puede tener una función idéntica a la del búfer 116, el filtro 320 de bucle puede tener una función idéntica a la del filtro 120 de bucle, y el búfer 330 de imágenes decodificadas puede tener una función idéntica a la del búfer 130 de imágenes decodificadas.

5 La unidad 360 de procesamiento de predicción puede comprender una unidad 344 de interpredicción y una unidad 354 de intrapredicción, en donde la unidad 344 de interpredicción puede parecerse a la unidad 144 de interpredicción en cuanto a la función, y la unidad 354 de intrapredicción puede parecerse a la unidad 154 de intrapredicción en cuanto a la 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
10 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.

15 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 en función de un modo de intrapredicción señalado y datos de bloques previamente decodificados del fotograma o imagen actual. Cuando la trama 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
20 para producir bloques 365 de predicción para un bloque de vídeo del segmento de vídeo actual en función de 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 tramas de referencia, Lista 0 y Lista 1, usando técnicas de construcción predeterminadas
25 en función de imágenes de referencia almacenadas en DPB 330.

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
30 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.
35

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
40 304 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.

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 para producir bloques residuales en el dominio de píxeles.
45

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
50 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,
55 un filtro bilateral o un filtro de bucle adaptativo (ALF) o un filtro de nitidez o filtros de suavizado o filtros colaborativos. Aunque la unidad 320 de filtro de bucle se muestra en la figura 3 como un filtro de bucle, en otras configuraciones, la unidad 320 de filtro de bucle puede implementarse como un filtro posterior al bucle.
60

Los bloques 321 de vídeo decodificados en un fotograma o una imagen dados se almacenan luego en el búfer 330 de imágenes decodificadas, que almacena imágenes de referencia usadas para la compensación de movimiento posterior.
65

El decodificador 30 está configurado para emitir la imagen 331 decodificada, por ejemplo, a través de la salida 332, para presentación o visualización a un usuario.

5 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 basado en transformación puede cuantificar de forma inversa la señal residual directamente sin la unidad 312 de procesamiento de transformación inversa para ciertos bloques o tramas. 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.

15 La figura 4 es un diagrama esquemático de un dispositivo 400 de codificación de vídeo según una realización de la divulgación. 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 codificador tal como el codificador 20 de vídeo de la figura 1A. En una realización, el dispositivo 400 de codificación de vídeo puede ser uno o más componentes del codificador 20 de vídeo de la figura 1A como se describió anteriormente.

20 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 central 430 de procesamiento (CPU, por sus siglas en inglés) para procesar los datos; unidades 440 de transmisión (Tx) 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.

30 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 de entrada 410, 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.

40 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 estática de acceso aleatorio (SRAM).

45 La figura 5 es un diagrama de bloques simplificado de un aparato 500 que se puede usar, bien como el dispositivo 310 de origen, bien como el dispositivo 320 de destino, o como ambos, de la figura 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, una tableta, un ordenador portátil, un portátil, un ordenador de escritorio, y similares.

50 Un procesador 502 en el aparato 500 puede ser una unidad central de procesamiento. 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.

60 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.

5 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.

10 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).

15 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 maneje 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

20 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.

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

25 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.

30 Aunque la figura 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 un 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

35 500. 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.

40 La codificación de vídeo como H.265/HEVC o H.266/VVC se basa en un marco híbrido de predicción más transformación. La unidad de árbol de codificación (CTU) se divide en una o más unidades de codificación (CU). Una CU puede contener un bloque de luminancia y dos bloques de crominancia (por ejemplo, formato YUV4:2:0 o YUV4:4:4), o una CU solo puede contener un bloque de luminancia, o una CU solo puede contener

45 dos bloques de crominancia. Una CU puede corresponder a una unidad de predicción (PU) y una unidad de transformación (TU), o puede corresponder a una unidad de predicción y múltiples unidades de transformación. La unidad de predicción contiene bloques de predicción de muestras de predicción y la unidad de transformación contiene bloques de transformación de muestras residuales. Las muestras reconstruidas de la CU se obtienen sumando las muestras de predicción con las muestras residuales correspondientes.

50 En el codificador, la información de predicción (por ejemplo, el modo de predicción y la información del vector de movimiento) y la información residual (por ejemplo, el modo de transformación, los coeficientes de transformación y el parámetro de cuantificación) de una CU se deciden mediante la optimización de la tasa de distorsión y luego se codifican en un flujo de bits. En el decodificador, se analizan la información de predicción

55 y la información residual. Según la información de predicción, se realiza una intrapredicción o interpredicción para generar las muestras de predicción de la unidad de predicción. Según la información residual, se realizan la cuantificación inversa y la transformación inversa para generar muestras residuales de la unidad de transformación. Las muestras de predicción y las muestras residuales se suman para obtener las muestras reconstruidas de la unidad de codificación.

60 Un modo de transformación básico es la residual de una CU que se transforma en una TU, es decir, el tamaño de la CU es igual al tamaño de la TU. En un árbol cuádruple residual (RQT) en HEVC, una CU se puede dividir en 4 TU de igual tamaño, y la información de si cada TU tiene residual o no se señala mediante indicadores de bloque codificados (cbf) de cada TU. La información de si la CU tiene residual está señalada por la raíz de cbf

65 de la CU (por ejemplo, el elemento de sintaxis `rqt_root_cbf` en HEVC).

La transformación central múltiple adaptativa (AMT) en VTM permite un bloque de transformación usando un tipo de transformación entre un conjunto de tipos de transformación. El tipo de transformación se señala en flujo de bits, y un tipo de transformación especifica la transformación horizontal (o transformación de fila) y la transformación vertical (o transformación de columna) del bloque de transformación. Las posibles transformaciones de núcleo para la transformación horizontal/vertical incluyen transformación de seno discreta (DST) y transformación de coseno discreta (DCT), por ejemplo, DST-7 (DST tipo 7), DST-1, DCT-8, DCT-2 y DCT5. En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, las posibles transformaciones de núcleo para la transformación horizontal/vertical incluyen la transformación de seno discreta (DST) y transformación de coseno discreta (DCT), por ejemplo, DST-4 (DST tipo 4), DST-1, DCT-4, DCT-2 y DCT5.

Para mejorar la eficacia de la codificación de transformación, en esta invención se introducen nuevos modos de transformación para una CU. Para cada uno de los nuevos modos de transformación, una CU se tesela en N (N=2, 3, o 4) TU, y solo una TU de la CU tiene residual y se infiere que el resto N-1 TU de la CU no tienen residual. Además, el tipo de transformación (es decir, la transformación horizontal y vertical) de la TU con residual se especifica mediante el modo de transformación. En otras palabras, cada uno de los nuevos modos de transformación especifica la partición de TU, la única TU con residual y el tipo de transformación de la TU con residual.

Una implementación de la presente divulgación describe el proceso de decodificación de una unidad de codificación. Se decodifica un flujo de bits que contiene al menos una imagen de datos de vídeo. Una imagen se divide en una pluralidad de regiones de imagen rectangulares y cada región corresponde a una unidad de árbol de codificación (CTU). Una CTU se divide en una pluralidad de unidades de codificación, según la información de partición de bloques contenida en el flujo de bits. La información de codificación de las unidades de codificación se analiza del flujo de bits, y los píxeles de las unidades de codificación se reconstruyen en función de la información de codificación. La información de codificación contiene información de predicción e información de transformación.

La información de predicción indica el modo de predicción de la CU (es decir, interpredicción o intrapredicción). Si el modo de predicción es intrapredicción, la información de predicción puede contener además un modo de intrapredicción, tal como el modo planar, el modo DC y el modo angular en HEVC y VVC. Si el modo de predicción es interpredicción, la información de predicción puede contener además si se usa el modo Omitir o el modo Combinar o el modo AMVP (predicción avanzada del vector de movimiento), el índice de candidato de combinación, la dirección de interpredicción, el índice de la trama de referencia, el vector de movimiento, si se usa la predicción basada en modelo afín, etc.

La información de transformación indica el modo de transformación de la CU y los coeficientes de transformación asociados con una o más TU en la CU. Además del modo de transformación convencional en el que el residual de una CU se transforma en una TU del mismo tamaño que el de la CU, esta realización también utiliza nuevos modos de transformación. Para cada uno de los nuevos modos de transformación, una CU se tesela en N (N=2, 3, o 4) TU, y solo una TU tiene residual y se infiere que el resto N-1 TU no tienen residual. Además, el tipo de transformación (es decir, la transformación horizontal y vertical) de la TU con residual se especifica mediante el modo de transformación.

En un ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos Q0, Q1, Q2 y Q3, como se ilustra en la figura 6. Para cada uno de los cuatro modos, una CU de tamaño W×H se divide en 4 TU de tamaño (W/2)×(H/2), es decir, la TU0 superior izquierda, la TU1 superior derecha, la TU2 inferior izquierda y la TU3 inferior derecha. Solo una de las TU0, TU1, TU2 y TU3 tiene residual.

1) Modo de transformación Q0: TU0 tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-8 y DCT-8, respectivamente.

2) Modo de transformación Q1: TU1 tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-7 y DCT-8, respectivamente.

3) Modo de transformación Q2: TU2 tiene residual; la transformación horizontal y la transformación vertical de TU2 son DCT-8 y DST-7, respectivamente.

4) Modo de transformación Q3: TU3 tiene residual; la transformación horizontal y la transformación vertical de TU3 son DST-7 y DST-7, respectivamente.

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en otro ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos Q0, Q1, Q2 y Q3 como se ilustra en la figura 6. Para cada uno de los cuatro modos de transformación, una CU de tamaño W×H se divide en 4 TU de tamaño (W/2)×(H/2), es decir la TU0 superior izquierda, la TU1 superior derecha, la TU2 inferior izquierda y la TU3

inferior derecha. Solo una de las TU0, TU1, TU2 y TU3 tiene residual.

- 5 1) Modo de transformación Q0: TU0 tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-4 y DCT-4, respectivamente.
- 2) Modo de transformación Q1: TU1 tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-4 y DCT-4, respectivamente.
- 10 3) Modo de transformación Q2: TU2 tiene residual; la transformación horizontal y la transformación vertical de TU2 son DCT-4 y DST-4, respectivamente.
- 4) Modo de transformación Q3: TU3 tiene residual; la transformación horizontal y la transformación vertical de TU3 son DST-4 y DST-4, respectivamente.

15 Para la TU que tiene residual, los indicadores de bloque codificados (cbf, por sus siglas en inglés) de tres componentes, es decir, los componentes Cb, Cr e Y, se pueden analizar a partir del flujo de bits, y el valor de al menos un cbf de los tres componentes es 1. Para cada una de las TU restantes sin residual, se infiere que los cbf de todos los componentes de color son 0, es decir, los cbf no se analizan a partir del flujo de bits.

20 En otro ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos HQ0, HQ1, VQ0 y VQ1, como se ilustra en la figura 7. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en dos TU de tamaño desigual, y solo la TU de menor tamaño tiene residual.

25 1) Modo de transformación HQ0: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/4)$ y TU1 de tamaño $W \times (3H/4)$; la parte superior TU0 tiene residual; la transformación horizontal de TU0 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2, y la transformación vertical de TU0 es DCT-8.

30 2) Modo de transformación HQ1: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (3H/4)$ y TU1 de tamaño $W \times (H/4)$; la TU1 inferior tiene residual; la transformación horizontal de TU1 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2, y la transformación vertical de TU1 es DST-7.

35 3) Modo de transformación VQ0: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/4) \times H$ y TU1 de tamaño $(3W/4) \times H$; la TU0 izquierda tiene residual; la transformación horizontal de TU0 es DCT-8 y la transformación vertical de TU0 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2.

40 4) Modo de transformación VQ1: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(3W/4) \times H$ y TU1 de tamaño $(W/4) \times H$; la TU1 derecha tiene residual; la transformación horizontal de TU1 es DST-7 y la transformación vertical de TU1 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2.

45 En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en otro ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos HQ0, HQ1, VQ0 y VQ1, como se ilustra en la figura 7. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en dos TU de tamaño desigual, y solo la TU de menor tamaño tiene residual.

50 1) Modo de transformación HQ0: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/4)$ y TU1 de tamaño $W \times (3H/4)$; la TU0 superior tiene residual; la transformación horizontal de TU0 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2, y la transformación vertical de TU0 es DCT-4. Como ejemplo, la transformación horizontal de TU0 es siempre DST-4 o siempre es DCT-2. Como otro ejemplo, la transformación horizontal de TU0 es DCT-2 si el ancho de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformación horizontal de TU0 es DST-4 si el ancho de TU0 es igual o menor que el umbral MaxMtsSize.

55

60 2) Modo de transformación HQ1: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (3H/4)$ y TU1 de tamaño $W \times (H/4)$; la TU1 inferior tiene residual; la transformación horizontal de TU1 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2, y la transformación vertical de TU1 es DST-4.

65 3) Modo de transformación VQ0: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/4) \times H$ y TU1 de tamaño $(3W/4) \times H$; la TU0 izquierda tiene residual; la transformación horizontal de TU0 es DCT-4 y la transformación vertical de TU0 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2. Como ejemplo, la transformación horizontal de TU0 es siempre DST-4 o siempre es DCT-2. Como otro

ejemplo, la transformación vertical de TU0 es DCT-2 si la altura de TU0 es mayor que el umbral MaxMtsSize, y la transformación vertical de TU0 es DST-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

- 4) Modo de transformación VQ1: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(3W/4) \times H$ y TU1 de tamaño $(W/4) \times H$; la TU1 derecha tiene residual; la transformación horizontal de TU1 es DST-4 y la transformación vertical de TU1 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2.

En un ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos HH0, HH1, VH0 y VH1, como se ilustra en la figura 8. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en dos TU de igual tamaño, y solo una TU tiene residual.

- 1) Modo de transformación HH0: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU0 superior tiene residual; la transformación horizontal de TU0 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2, y la transformación vertical de TU0 es DCT-8.

- 2) Modo de transformación HH1: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU1 inferior tiene residual; la transformación horizontal de TU1 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2, y la transformación vertical de TU1 es DST-7.

- 3) Modo de transformación VH0: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU0 izquierda tiene residual; la transformación horizontal de TU0 es DCT-8 y la transformación vertical de TU0 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2.

- 4) Modo de transformación VH1: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU1 derecha tiene residual; la transformación horizontal de TU1 es DST-7 y la transformación vertical de TU1 es una transformación central predefinida, por ejemplo, DST-7 o DCT-2.

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en un ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos HH0, HH1, VH0 y VH1, como se ilustra en la figura 8. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en dos TU de igual tamaño, y solo una TU tiene residual.

- 1) Modo de transformación HH0: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU0 superior tiene residual; la transformación horizontal de TU0 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2, y la transformación vertical de TU0 es DCT-4. Como ejemplo, la transformación horizontal de TU0 es siempre DST-4 o siempre es DCT-2. Como otro ejemplo, la transformación horizontal de TU0 es DCT-2 si el ancho de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformación horizontal de TU0 es DST-4 si el ancho de TU0 es igual o menor que el umbral MaxMtsSize.

- 2) Modo de transformación HH1: la CU se divide mediante una división horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU1 inferior tiene residual; la transformación horizontal de TU1 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2, y la transformación vertical de TU1 es DST-4.

- 3) Modo de transformación VH0: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU0 izquierda tiene residual; la transformación horizontal de TU0 es DCT-4 y la transformación vertical de TU0 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2. Como ejemplo, la transformación horizontal de TU0 es siempre DST-4 o es siempre DCT-2. Como otro ejemplo, la transformación vertical de TU0 es DCT-2 si la altura de TU0 es mayor que el umbral MaxMtsSize, y la transformación vertical de TU0 es DST-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

- 4) Modo de transformación VH1: la CU se divide mediante una división vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU1 derecha tiene residual; la transformación horizontal de TU1 es DST-4 y la transformación vertical de TU1 es una transformación central predefinida, por ejemplo, DST-4 o DCT-2.

En otro ejemplo, los nuevos modos de transformación incluyen los siguientes dos modos HH2 y VH2, como se ilustra en la figura 9. Para cada uno de los dos modos, una CU de tamaño $W \times H$ se divide en tres TU, y solo la TU de la mitad del tamaño de la CU tiene residual.

1) Modo de transformación HH2: la CU se divide en dirección horizontal en tres TU, TU0 de tamaño $W \times (H/4)$, TU1 de tamaño $W \times (H/2)$, TU2 de tamaño $W \times (H/4)$; la TU1 central tiene residual; la transformación horizontal y la transformación vertical de TU0 son DST-7 y DCT-2 (o DST-7 y DST-1), respectivamente.

5

2) Modo de transformación VH2: la CU se divide en dirección vertical en tres TU, TU0 de tamaño $(W/4) \times H$, TU1 de tamaño $(W/2) \times H$, y TU2 de tamaño $(W/4) \times H$; la TU1 central tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-2 y DST-7 (o DST-1 y DST-7), respectivamente.

10 En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en otro ejemplo, los nuevos modos de transformación incluyen los siguientes dos modos HH2 y VH2, como se ilustra en la figura 9. Para cada uno de los dos modos, una CU de tamaño $W \times H$ se divide en tres TU, y solo la TU de la mitad del tamaño de la CU tiene residual.

15

1) Modo de transformación HH2: la CU se divide en dirección horizontal en tres TU, TU0 de tamaño $W \times (H/4)$, TU1 de tamaño $W \times (H/2)$, TU2 de tamaño $W \times (H/4)$; la TU1 central tiene residual; la transformación horizontal y la transformación vertical de TU0 son DST-4 y DCT-2 (o DST-4 y DST-1), respectivamente.

20 2) Modo de transformación VH2: la CU se divide en dirección vertical en tres TU, TU0 de tamaño $(W/4) \times H$, TU1 de tamaño $(W/2) \times H$, y TU2 de tamaño $(W/4) \times H$; la TU1 central tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-2 y DST-4 (o DST-1 y DST-4), respectivamente.

25 En otro ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos QH0, QH1, QH2 y QH3, como se ilustra en la figura 10. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en 3 TU, dos de las cuales son de tamaño $(W/2) \times (H/2)$, y la otra es de tamaño $W \times (H/2)$, es decir, TU0, TU1 y TU2. Solo una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual.

30 1) Modo de transformación QH0: la TU0 superior izquierda tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-8 y DCT-8, respectivamente.

2) Modo de transformación QH1: la TU1 superior derecha tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-7 y DCT-8, respectivamente.

35 3) Modo de transformación QH2: la TU1 inferior izquierda tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-8 y DST-7, respectivamente.

4) Modo de transformación QH3: la TU2 inferior derecha tiene residual; la transformación horizontal y la transformación vertical de TU2 son DST-7 y DST-7, respectivamente.

40

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en otro ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos QH0, QH1, QH2 y QH3, como se ilustra en la figura 10. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en 3 TU, dos de las cuales son de tamaño $(W/2) \times (H/2)$, y la otra es de tamaño $W \times (H/2)$, es decir, TU0, TU1 y TU2. Solo una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual.

45

1) Modo de transformación QH0: la TU0 superior izquierda tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-4 y DCT-4, respectivamente.

50

2) Modo de transformación QH1: la TU1 superior derecha tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-4 y DCT-4, respectivamente.

3) Modo de transformación QH2: la TU1 inferior izquierda tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-4 y DST-4, respectivamente.

55

4) Modo de transformación QH3: la TU2 inferior derecha tiene residual; la transformación horizontal y la transformación vertical de TU2 son DST-4 y DST-4, respectivamente.

60 En un ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos QV0, QV1, QV2 y QV3, como se ilustra en la figura 11. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en 3 TU, dos de las cuales son de tamaño $(W/2) \times (H/2)$, y la otra es de tamaño $(W/2) \times H$, es decir, TU0, TU1 y TU2. Solo una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual.

65 1) Modo de transformación QV0: la TU0 superior izquierda tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-8 y DCT-8, respectivamente.

2) Modo de transformación QV1: la TU1 superior derecha tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-7 y DCT-8, respectivamente.

5 3) Modo de transformación QV2: la TU1 inferior izquierda tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-8 y DST-7, respectivamente.

4) Modo de transformación QV3: la TU2 inferior derecha tiene residual; la transformación horizontal y la transformación vertical de TU2 son DST-7 y DST-7, respectivamente.

10

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en un ejemplo, los nuevos modos de transformación incluyen los siguientes cuatro modos QV0, QV1, QV2 y QV3, como se ilustra en la figura 11. Para cada uno de los cuatro modos, una CU de tamaño $W \times H$ se divide en 3 TU, dos de las cuales son de tamaño $(W/2) \times (H/2)$, y la otra es de tamaño $(W/2) \times H$, es decir, TU0, TU1 y TU2. Solo una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual.

15

1) Modo de transformación QV0: la TU0 superior izquierda tiene residual, la transformación horizontal y la transformación vertical de TU0 son DCT-4 y DCT-4, respectivamente.

20

2) Modo de transformación QV1: la TU1 superior derecha tiene residual; la transformación horizontal y la transformación vertical de TU1 son DST-4 y DCT-4, respectivamente.

3) Modo de transformación QV2: la TU1 inferior izquierda tiene residual; la transformación horizontal y la transformación vertical de TU1 son DCT-4 y DST-4, respectivamente.

25

4) Modo de transformación QV3: la TU2 inferior derecha tiene residual; la transformación horizontal y la transformación vertical de TU2 son DST-4 y DST-4, respectivamente.

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 6 modos entre los modos mencionados anteriormente: HH0, HH1, HH2, VH0, VH1 y VH2. Se puede usar un indicador de nivel de segmento para señalar si los 6 modos de transformación están permitidos para un segmento en particular. Donde los nuevos modos de transformación permitidos para las CU significa que cuando se decodifica una CU, el modo de transformación de destino para la TU residual de la CU se puede obtener entre los nuevos modos de transformación permitidos para las CU. Los nuevos modos de transformación permitidos para CU pueden ser un subconjunto de los 22 modos de transformación mencionados anteriormente, por lo que pueden reducir los modos de transformación candidatos para el modo de transformación de destino, para mejorar la eficiencia de decodificación y reducir los bits necesarios para transmitir el índice/índices para indicar el modo de transformación de destino, para ahorrar el ancho de banda necesario para transmitir los datos de vídeo codificados. Cabe señalar que, los nuevos modos de transformación permitidos para CU pueden predefinirse/preestablecerse o determinarse en tiempo real, cuando los nuevos modos de transformación permitidos para las CU se predefinen/preestablecen/determinan, el decodificador sabrá qué nuevos modos de transformación son modos de transformación candidatos para la TU residual de la CU.

30

35

40

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 8 modos entre los modos mencionados anteriormente: Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0 y VQ1.

45

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 8 modos entre los modos mencionados anteriormente: QH0, QH1, QH2, QH3, HQ0, HQ1, VQ0 y VQ1.

50

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 8 modos entre los modos mencionados anteriormente: QV0, QV1, QV2, QV3, HQ0, HQ1, VQ0 y VQ1.

En un ejemplo según la invención reivindicada, los nuevos modos de transformación permitidos para las CU incluyen 8 modos entre los modos de transformación mencionados anteriormente: HH0, HH1, VH0, VH1, HQ0, HQ1, VQ0 y VQ1.

55

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 12 modos entre los modos mencionados anteriormente: Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0, VQ1, HH0, HH1, VH0 y VH1.

60

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 14 modos entre los modos mencionados anteriormente: Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0, VQ1, HH0, HH1, HH2, VH0, VH1 y VH2.

En un ejemplo, los nuevos modos de transformación permitidos para las CU incluyen 14 modos entre los modos mencionados anteriormente: QH0, QH1, QH2, QH3, HQ0, HQ1, VQ0, VQ1, HH0, HH1, HH2, VH0, VH1 y VH2.

65

En un ejemplo, los nuevos modos de transformación permitidos para una pluralidad de CU en un segmento se señalan en el conjunto de parámetros de secuencia (SPS) o en la cabecera del segmento. Los modos HH0, HH1, HH2, VH0, VH1 y VH2 se agrupan en un primer grupo. Los modos Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0 y VQ1 se agrupan en un segundo grupo. La habilitación de cada grupo se señala mediante un indicador, es decir, se señala un primer indicador para indicar si se permite el primer grupo de modos y se señala un segundo indicador para indicar si se permite el segundo grupo de modos.

En un ejemplo, los nuevos modos de transformación permitidos para una pluralidad de CU en un segmento se señalan en el conjunto de parámetros de secuencia o en la cabecera de segmento del segmento. Los modos HH0, HH1, VH0 y VH1 se agrupan en un primer grupo. Los modos Q0, Q1, Q2 y Q3 se agrupan en un segundo grupo. Los modos HQ0, HQ1, VQ0 y VQ1 se agrupan en un tercer grupo. Se señala un índice para indicar cuál de los tres grupos está permitido. El valor de índice 0 indica que no se permiten todos los tres grupos, el valor de índice 1 indica que solo se permite el primer grupo, el valor de índice 2 indica que se permite el segundo grupo, el valor de índice 3 indica que se permite el tercer grupo, el valor de índice 4 indica que se permiten el segundo y el tercer grupos, y el valor de índice 5 indica que se permiten todos los tres grupos.

Los nuevos modos de transformación pueden aplicarse a todas las CU o aplicarse a las CU según la información de predicción de la CU. En un ejemplo, los nuevos modos de transformación se aplican solo a las CU intrapredichas. En un ejemplo, los nuevos modos de transformación se aplican solo a las CU interpredichas. En otro ejemplo, los nuevos modos de transformación pueden permitirse a las CU que usan un método de interpredicción específico (por ejemplo, compensación de movimiento basada en el modelo de traslación) pero no se permite a las CU que usan otro método de interpredicción (por ejemplo, compensación de movimiento basada en un modelo afín). En otro ejemplo, los nuevos modos de transformación pueden permitirse para las CU predichas por el modo de combinación o por el modo AMVP (predicción avanzada del vector de movimiento) con precisión de diferencia de vector de movimiento de 1/4-pel, pero no se permite para las CU que usan el modo de combinación afín, el modo interafín o el modo AMVP con precisión de diferencia de vector de movimiento de 1-pel o 4-pel. En otro ejemplo, los nuevos modos de transformación pueden permitirse para las CU que usan el modo de combinación con un índice de combinación menor que 2, pero no se permite para las CU que usan el modo de combinación con un índice de combinación no menor que 2. En el estándar H.265/HEVC se hace referencia al modo de combinación y el modo AMVP. En el códec JEM (modelo de escaneo conjunto) de JVET (equipo de escaneo de vídeo conjunto) se puede hacer referencia al modo de combinación afín y al modo interafín.

Los nuevos modos de transformación pueden permitirse según el tamaño de la CU. En un ejemplo, si una CU que está siendo tileada por uno de los modos de transformación da como resultado una TU con un lado menor que un umbral (por ejemplo, 4, 8 o 16 píxeles de luminancia, etc.), el modo de transformación no está permitido para la CU. En un ejemplo, si una CU que está siendo tileada por uno de los modos de transformación da como resultado una TU con un lado menor que un umbral (por ejemplo, 32, o 64, o 128 píxeles de luminancia, etc.), el modo de transformación no está permitido para la CU. En un ejemplo, una CU con un lado (es decir, ancho o alto) más grande que un umbral (por ejemplo, 32 o 64 píxeles de luminancia) no puede usar los nuevos modos de transformación.

La figura 12 ilustra un método de decodificación que incluye:

1201. analizar un flujo de bits recibido para obtener información de predicción de una CU;

Este es un proceso comúnmente usado en la decodificación de vídeo, tal como la unidad de codificación de análisis y la estructura de sintaxis de la unidad de predicción en H.265/HEVC o H.266/VVC.

1202. obtener un modo de transformación de destino de la TU residual cuando la CU tiene solo una unidad de transformación (TU) residual y el tamaño de la TU residual es menor que el tamaño de la CU; en donde el modo de transformación de destino especifica un modo de partición de TU de la CU, la posición de la TU residual (por ejemplo, descrita por el índice de la TU residual entre todas las TU de la CU, o descrita por las coordenadas de la esquina superior izquierda de la TU residual y el ancho y alto de la TU residual), y un tipo de transformación de la TU residual;

El modo de transformación de destino es uno cualquiera de los modos mencionados anteriormente: Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0, VQ1, HH0, HH1, VH0, VH1, HH2, VH2, QH0, QH1, QH2, QH3, QV0, QV1, QV2 y QV3.

En algunas formas de implementación, para mejorar la eficacia de la decodificación, primero se comprueba la presencia de información de la unidad de transformación en la CU. Donde la presencia de información de la unidad de transformación puede indicarse mediante un elemento de sintaxis cbf raíz. El cbf raíz igual a 0 indica que la información de la unidad de transformación de la CU no está presente en el flujo de bits (es decir, la CU no tiene residual), mientras que cbf raíz igual a 1 indica que la información de la unidad de transformación de la CU está presente en el flujo de bits. Para una CU interpredicha, la cbf raíz igual a 1 implica que la CU tiene

residual. La etapa 1102 se realiza cuando la CU tiene información de la unidad de transformación. Cuando la CU no tiene información de unidad de transformación, la residual de la CU se establece como cero y se realiza la etapa 1105 directamente.

5 1203. analizar el flujo de bits recibido para obtener coeficientes de transformación de la TU residual;

Esto es un proceso comúnmente usado en la decodificación de vídeo.

10 1204. aplicar una cuantificación inversa a los coeficientes de transformación de la TU residual para obtener coeficientes decuantificados.

Esto es un proceso comúnmente usado en la decodificación de vídeo.

15 1205. Aplicar, en función del modo de transformación de destino, una transformación inversa a los coeficientes decuantificados para obtener un bloque residual de la TU residual.

Además de la TU residual, las residuales de las demás TU de la CU se establecen a cero.

20 1206. obtener un bloque de predicción de la CU en función de la información de predicción;

Esto es un proceso comúnmente usado en la decodificación de vídeo, tal como la intrapredicción y la interpredicción en HEVC y VVC.

25 1207. obtener un bloque de vídeo en función del bloque residual y del bloque de predicción;

En una forma de implementación, el bloque de vídeo se obtiene sumando el residual de la CU con la predicción de la CU.

30 1208. emitir una secuencia de vídeo, incluyendo la secuencia de vídeo un fotograma de vídeo que incluye el bloque de vídeo.

Esto es un proceso comúnmente usado en la decodificación de vídeo.

35 En una forma de implementación, obtener el modo de transformación de destino de la TU residual comprende: analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual; y obtener el modo de transformación de destino de la TU residual en función del índice de modo. Por ejemplo, si se permite en un segmento el uso de 8 modos entre los 22 modos de transformación de destino mencionados anteriormente, el índice de modo puede indicarse mediante un índice de modo con M contenedores (por ejemplo, M = 3, 4 o 5), es decir, el índice de modo se binariza en M contenedores en el proceso de codificación aritmética binaria adaptativa basada en contexto (CABAC) (donde M puede ser diferente para diferentes esquemas de binarización), o el índice de modo puede expresarse por M indicadores.

40 Para reducir el consumo de bits del índice de modo, en otra forma de implementación, antes de analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual, el método comprende además: determinar los modos de transformación de destino candidatos de la TU residual en función de un tamaño de la CU; donde el análisis del flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual comprende: analizar, en función de los modos de transformación de destino candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual. Por lo tanto, dependiendo del tamaño de la CU, la cantidad de modos de transformación de destino candidatos puede ser inferior a 8, por lo que se pueden guardar 1 o más contenedores del indicador de modo.

45 Los modos de transformación de destino candidatos pueden ser función del tamaño de la CU. Por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los modos de transformación de destino candidatos pueden incluir el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3.

50 Por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos pueden incluir el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

55 por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los modos de transformación de destino candidatos pueden incluir el modo de transformación VH0 y el modo de transformación VH1;

60 por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y un ancho de la CU está en el

intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

5 por ejemplo, cuando una altura de la CU está en el intervalo de [Th1×2, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación HH0 y el modo de transformación HH1;

10 por ejemplo, cuando un ancho de la CU está en el intervalo de [Th1×2, Th2] y una altura de la CU está en el intervalo de [Th1×2, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

15 por ejemplo, cuando un ancho de la CU está en el intervalo de [Th1×4, Th2] y una altura de la CU está en el intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

20 por ejemplo, cuando un ancho de la CU está en el intervalo de [Th1×2, Th2] y una altura de la CU está en el intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación VH0 y el modo de transformación VH1;

por ejemplo, cuando una altura de la CU está en el intervalo de [Th1×4, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

25 por ejemplo, cuando una altura de la CU está en el intervalo de [Th1×2, Th2] y un ancho de la CU está en el intervalo de [Th1, Th2], los modos de transformación de destino candidatos pueden incluir el modo de transformación HH0 y el modo de transformación HH1.

30 Th1, Th2 y Th3 son valores enteros predefinidos, por ejemplo, Th1 es 4, 8 o 16; Th2 es 32, 64 o 128; y Th3 es 64, 128 o 256.

En otra forma de implementación, para reducir el consumo de bits para indicar el modo de transformación de destino, los 22 modos mencionados anteriormente se agrupan en grupos de modos. Por ejemplo, el grupo de modos puede incluir al menos un grupo de la siguiente manera:

35 un grupo de modos incluye el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

40 un grupo de modos incluye el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

45 un grupo de modos incluye el modo de transformación HH0, HH1, HH2, VH0, VH1 y VH2;

un grupo de modos incluye el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

50 un grupo de modos incluye el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

55 un grupo de modos incluye el modo de transformación QV0, el modo de transformación QV1, el modo de transformación QV2, el modo de transformación QV3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

60 un grupo de modos incluye el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

65 un grupo de modos incluye el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2, el modo de transformación Q3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de

transformación VH1 y el modo de transformación VH2;

5 un grupo de modos incluye el modo de transformación QH0, el modo de transformación QH1, el modo de transformación QH2, el modo de transformación QH3, el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0, el modo de transformación VQ1, el modo de transformación HH0, el modo de transformación HH1, el modo de transformación HH2, el modo de transformación VH0, el modo de transformación VH1 y el modo de transformación VH2;

10 un grupo de modos incluye el modo de transformación HH0, el modo de transformación HH1, el modo de transformación VH0 y el modo de transformación VH1;

un grupo de modos incluye el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3; o

15 un grupo de modos incluye el modo de transformación HQ0, el modo de transformación HQ1, el modo de transformación VQ0 y el modo de transformación VQ1;

20 donde la etapa de obtener el modo de transformación de destino de la TU residual puede incluir: analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el grupo de transformación de destino; analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino en el grupo de modos; y obtener el modo de transformación de destino en función del índice de grupo y el índice de modo.

25 En una forma de implementación, antes de analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino, el método comprende además: determinar al menos un grupo de modos candidatos de la TU residual en función del tamaño de la CU; y correspondientemente, analizar, en función del al menos un grupo de modos candidatos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino.

30 Por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th3]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th3]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

35 por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

40 por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

45 por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2;

50 por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1;

55 por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación Q0, el modo de transformación Q1, el modo de transformación Q2 y el modo de transformación Q3;

60 por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VQ0, el modo de transformación VQ1 y el modo de transformación VH2;

por ejemplo, cuando un ancho de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una altura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación VH0 y el modo de transformación VH1;

65 por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el

modo de transformación HQ0, el modo de transformación HQ1 y el modo de transformación HH2; o

5 por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y un ancho de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende el modo de transformación HH0 y el modo de transformación HH1.

Los umbrales Th1, Th2 y Th3 son valores enteros predefinidos. Por ejemplo, Th1 es 4, 8 o 16; Th2 es 32, 64 o 128; Th3 es 64, 128 o 256.

10 En otra forma de implementación, antes de obtener el modo de transformación de destino de la TU residual, el método comprende además: determinar, en función de la información de predicción de la CU, si se obtiene el modo de transformación de destino de la TU residual.

15 Por ejemplo, cuando un modo de predicción de la CU es un modo de intrapredicción, determinar obtener el modo de transformación de destino de la TU residual;

Por ejemplo, cuando un modo de predicción de la CU es un modo de interpredicción, determinar obtener el modo de transformación de destino de la TU residual;

20 Por ejemplo, cuando un método de predicción de la CU es un método de predicción positiva preestablecido, determinar obtener el modo de transformación de destino de la TU residual. El método de predicción positiva preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo de traslación; un método de predicción de combinación; un método avanzado de predicción de vectores de movimiento con precisión de diferencia de vector de movimiento de 1/4-pel; o un método de predicción de combinación con un índice de combinación menor que 2.

30 Por ejemplo, cuando un método de predicción de la CU es un método de predicción negativa preestablecido, determinar no obtener el modo de transformación de destino de la TU residual. El método de predicción negativa preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en un modelo afín; un método de predicción de combinación afín; un método de interpredicción afín; un método avanzado de predicción del vector de movimiento con precisión de diferencia de vector de movimiento de 1-pel o 4-pel; o un método de predicción de combinación con un índice de combinación no menor que 2.

35 En otra forma de implementación, antes de obtener el modo de transformación de destino de la TU residual, el método comprende además: determinar, en función de un tamaño de la CU, si obtener el modo de transformación de destino de la TU residual.

40 Por ejemplo, cuando un ancho o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el modo de transformación de destino de la TU residual. El umbral de CU preestablecido puede ser de 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia.

45 Por ejemplo, cuando un ancho o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual. El umbral de TU mínimo preestablecido puede ser de 4 píxeles de luminancia, 8 píxeles de luminancia o 16 píxeles de luminancia.

50 Por ejemplo, cuando un ancho o una altura de una TU de la CU es mayor que un umbral de TU máximo preestablecido, determinar no obtener el modo de transformación de destino de la TU residual. El umbral de TU preestablecido es de 32 píxeles de luminancia, 64 píxeles de luminancia o 128 píxeles de luminancia.

55 Cabe señalar que, incluso según la información de predicción de la CU y/o el tamaño de la CU, determinar obtener el modo de transformación de destino de la TU residual, solo significa que se puede obtener el modo de transformación de destino, pero no significa que el modo de transformación de destino vaya a obtenerse realmente. Porque en el lado del codificador, la elección del modo de transformación para hacer la transformación depende de muchos factores. Es decir, los nuevos modos de transformación mencionados anteriormente (los 22 modos de transformación mencionados anteriormente) se añaden como candidatos, los modos de transformación ya existentes (antiguos modos de transformación), por ejemplo, la CU se transforma en una TU del tamaño de la CU, pueden seguir usándose.

60 Por lo tanto, basándose en que la CU tiene información de la unidad de transformación, cuando se determina, según la información de predicción de la CU y/o el tamaño de la CU, para obtener el modo de transformación de destino de la TU residual; analizar el flujo de bits puede obtener un indicador de modo que indica los modos de transformación antiguos. En consecuencia, cuando se determina no obtener el modo de transformación de destino de la TU residual, lo que significa que se usan los modos de transformación antiguos para la CU. Si la CU es transformada por una TU del tamaño de la CU, la TU residual es del tamaño de la CU, y el tipo de

transformación de la TU residual puede ser DCT-2 para transformación horizontal y DCT-2 para transformación vertical por defecto o puede señalarse en flujo de bits (por ejemplo, el método AMT en el *software* VTM, en el que se permiten múltiples tipos de transformación para una TU y se señala el tipo de transformación usado por la TU).

5

Por ejemplo, si los modos de transformación de Q0, Q1, Q2, Q3, VQ0, VQ1, HQ0, HQ1, VH0, VH1, VH2, HH0, HH1 y HH2 son modos de transformación de destino candidatos de la CU, el decodificador puede analizar primero el flujo de bits para determinar si la CU usa uno de los modos de transformación de destino candidatos o si la CU es transformada por una TU del tamaño de la CU. Por ejemplo, se puede usar un indicador para

10

indicar si la CU usa uno de los modos de transformación de destino candidatos o si la CU es transformada por una TU del tamaño de la CU.

Si la CU usa uno de los modos de transformación de destino candidatos, analizar el flujo de bits para obtener el modo de transformación de la CU. En un método, Q0, Q1, Q2 y Q3 se agrupan en el grupo A, VQ0 y VQ1 se agrupan en el grupo B, HQ0 y HQ1 se agrupan en el grupo C, VH0, VH1 y VH2 se agrupan en el grupo D, y HH0, HH1 y HH2 se agrupan en el grupo E. Primero se decide el grupo al que pertenece el modo de transformación de la CU analizando un índice de grupo y luego se decide el modo de transformación de destino dentro del grupo analizando otro índice.

15

20

El análisis del índice de grupo puede basarse en qué grupos de modos se determinan/preestablecen como grupos de modos candidatos. Por ejemplo, cuando el grupo A y cualquier otro grupo son grupos de modos candidatos, se analiza un primer indicador para determinar si el modo de transformación de destino pertenece al grupo A o no; de lo contrario, si el primer indicador no está presente en el flujo de bits, y se infiere que el modo de transformación pertenece al grupo A si el grupo A es un grupo de modos candidatos, o se infiere que pertenece a los grupos B, C, D y E si el grupo A no es un grupo de modos candidatos. Si el modo de transformación de destino pertenece a los grupos B, C, D y E, se analiza un segundo indicador para determinar si el modo de transformación de destino pertenece a B/D (es decir, grupo B o D) o C/E si ambos grupos B/D y C/E son candidatos, y el segundo indicador se puede inferir si solo B/D es candidato o si solo C/E es candidato. Si el modo de transformación pertenece a B/D, se puede analizar un tercer indicador para determinar si el modo de transformación de destino pertenece a B. De manera similar, si el modo de transformación de destino pertenece a C/E, el tercer indicador se puede analizar para determinar si el modo de transformación pertenece a C.

25

30

35

De forma similar al análisis del índice de grupo, el análisis del índice del modo de transformación de destino dentro de un grupo puede basarse en qué modos de transformación hay en el grupo.

40

Por ejemplo, si los modos de transformación Q0, Q1, Q2, Q3, VQ0, VQ1, HQ0 y HQ1 son modos de transformación de destino candidatos de la CU, el decodificador puede analizar primero el flujo de bits para determinar si la CU usa uno de los modos de transformación candidatos o si la CU es transformada por una TU del tamaño de la CU. Por ejemplo, se puede usar un indicador para indicar si la CU usa uno de los modos de transformación de destino candidatos o si la CU es transformada por una TU del tamaño de la CU.

45

Si la CU usa uno de los modos de transformación de destino candidatos, analizar el flujo de bits para obtener el modo de transformación de la CU. En un método, los modos de transformación Q0, Q1, Q2 y Q3 se agrupan en el grupo A, VQ0 y VQ1 se agrupan en el grupo B, y HQ0 y HQ1 se agrupan en el grupo C. Primero se decide el grupo al que pertenece el modo de transformación de destino, analizando un índice de grupo, y luego se decide el modo de transformación de destino dentro del grupo analizando un índice de modo.

50

La presente divulgación divulga además un decodificador de vídeo, la estructura del decodificador se ilustra en la figura 3, el decodificador de vídeo incluye:

55

una unidad 304 de decodificación de entropía, configurada para analizar un flujo de bits recibido para obtener información de predicción de una unidad de codificación (CU); obtener un modo de transformación de destino de una unidad de transformación (TU) residual cuando la TU residual es la única TU residual de la CU y un tamaño de la TU residual es menor que un tamaño de la CU, en donde el modo de transformación de destino especifica un modo de partición de TU de la CU, la posición de la TU residual y un tipo de transformación de la TU residual; y analizar el flujo de bits recibido para obtener los coeficientes de transformación de la TU residual. Donde el modo de transformación de destino es el mismo que se mencionó anteriormente.

60

una unidad 310 de procesamiento de cuantificación inversa, configurada para aplicar una cuantificación inversa a los coeficientes de transformación de la TU residual para obtener coeficientes decuantificados;

65

una unidad 312 de procesamiento de transformación inversa, configurada para aplicar una transformación inversa a los coeficientes decuantificados en función del modo de transformación de destino para obtener un bloque residual de la TU residual;

una unidad 360 de procesamiento de predicción, configurada para obtener un bloque de predicción de la CU en función de la información de predicción;

5 una unidad 314 de reconstrucción, configurada para obtener un bloque de vídeo en función del bloque residual y del bloque de predicción; y

una salida 332, configurada para emitir una secuencia de vídeo, incluyendo la secuencia de vídeo una trama de vídeo que incluye el bloque de vídeo.

10 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual; y obtener el modo de transformación de destino de la TU residual en función del índice de modo.

15 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para determinar los modos de transformación de destino candidatos de la TU residual en función del tamaño de la CU; y

20 analizar, en función de los modos de transformación de destino candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino de la TU residual.

En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para determinar modos de transformación de destino candidatos de la TU residual según el método indicado anteriormente.

25 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para analizar el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino; y analizar el flujo de bits para obtener un índice de modo para indicar el modo de transformación de destino en el grupo de modos; y

30 obtener el modo de transformación de destino en función del índice de grupo y el índice de modo.

Donde el modo de transformación de destino pertenece a cualquier grupo de modos como se indicó anteriormente.

35 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para determinar al menos un grupo de modos candidatos de la TU residual en función del tamaño de la CU; y

40 analizar, en función del al menos un grupo de modos candidatos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el modo de transformación de destino.

Donde la unidad de decodificación de entropía está configurada además para determinar al menos un grupo de modos candidatos según al menos uno de los métodos indicados anteriormente.

45 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para determinar, en función de la información de predicción de la CU, si se obtiene el modo de transformación de destino de la TU residual.

50 Donde la unidad 304 de decodificación de entropía está configurada además para determinar si se obtiene el modo de transformación de destino de la TU residual según al menos uno de los métodos como se indicó anteriormente.

55 En una forma de implementación, la unidad 304 de decodificación de entropía está configurada además para determinar, en función de un tamaño de la CU, si se obtiene el modo de transformación de destino de la TU residual.

Donde la unidad 304 de decodificación de entropía está configurada además para determinar si se obtiene el modo de transformación de destino de la TU residual según al menos uno de los métodos como se indicó anteriormente.

60 La presente descripción describe además un decodificador que comprende un circuito de procesamiento para llevar a cabo el método como se indicó anteriormente.

65 La presente descripción describe además un producto de programa informático que comprende un código de programa para realizar el método como se indicó anteriormente.

La presente descripción describe además un decodificador, que comprende:

uno o más procesadores; y

5 un medio de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena la 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 como se indicó anteriormente.

10 La figura 13 ilustra un ejemplo de intrapredicción 1300 en codificación de vídeo que emplea modos de intrapredicción. Como se muestra, un bloque 1301 actual puede predecirse mediante muestras en bloques 1310 vecinos. Los codificadores generalmente pueden codificar una imagen de la parte superior izquierda a la parte inferior derecha. Sin embargo, los codificadores pueden codificar de derecha a izquierda en algunos casos, como se analiza a continuación. Cabe señalar que, como se usa en la presente memoria, la derecha se refiere al lado derecho de una imagen codificada, la izquierda se refiere al lado izquierdo de una imagen codificada, la parte superior se refiere al lado superior de una imagen codificada y la parte inferior se refiere al lado inferior de una imagen codificada.

15 Cabe señalar que un bloque 1301 actual puede no coincidir siempre exactamente con una muestra de un bloque 1310 vecino. En tal caso, se codifica un modo de predicción a partir del bloque 1310 vecino de coincidencia más cercana. Para permitir que el decodificador determine el valor adecuado, se conserva una diferencia entre el valor predicho y el valor real. Esto se conoce como información residual. La información residual se produce tanto en la intrapredicción 1300 como en la interpredicción.

20 La figura 14 es un diagrama esquemático de un mecanismo 1400 de codificación de vídeo ejemplar, basado en la intrapredicción 1300 y/o interpredicción. Un bloque 1401 de imagen puede obtenerse mediante un codificador a partir de una o más tramas. Por ejemplo, una imagen puede dividirse en una pluralidad de regiones de imagen rectangulares. Cada región de la imagen corresponde a una unidad de árbol de codificación (CTU). Una CTU se divide en una pluralidad de bloques, tales como las unidades de codificación en HEVC. Luego, la información de partición de bloque se codifica en un flujo de bits 1411. En consecuencia, el bloque 301 de imagen es una parte dividida de una imagen y contiene píxeles que representan componentes de luminancia y/o componentes de crominancia en una parte correspondiente de la imagen. Durante la codificación, el bloque 1401 de imagen se codifica como un bloque 303 de predicción que contiene información de predicción, tal como modos de predicción para la intrapredicción y/o vectores de movimiento para la interpredicción. Codificar el bloque 1401 de imagen como un bloque 1403 de predicción puede dejar entonces un bloque 1405 residual que contiene información residual que indica la diferencia entre el bloque 1403 de predicción y el bloque 1401 de imagen.

35 Cabe señalar que un bloque 1401 de imagen puede dividirse como una unidad de codificación que contiene un bloque 1403 de predicción y un bloque 1405 residual. El bloque 1403 de predicción puede contener todas las muestras de predicción de la unidad de codificación y el bloque 1405 residual puede contener todas las muestras residuales de la unidad de codificación. En tal caso, el bloque 1403 de predicción tiene el mismo tamaño que el bloque 1405 residual. En otro ejemplo, el bloque 1401 de imagen se puede dividir como una unidad de codificación que contiene dos bloques 1403 de predicción y un bloque 1405 residual. En tal caso, cada bloque 1405 de predicción contiene una parte de las muestras de predicción de la unidad de codificación, y el bloque 1405 residual contiene todas las muestras residuales de la unidad de codificación. En otro ejemplo, el bloque 1401 de imagen se puede dividir como una unidad de codificación que contiene dos bloques 1403 de predicción y cuatro bloques 1405 residuales. El patrón de partición de los bloques 1405 residuales en una unidad de codificación puede señalarse en el flujo de bits 1411. Tales patrones de posición pueden incluir un árbol cuádruple residual (RQT) en HEVC. Además, un bloque 1401 de imagen puede contener solo componentes de luminancia (por ejemplo, luz), indicados como componentes Y, de muestras de imagen (o píxeles). En otros casos, el bloque 1401 de imagen puede contener los componentes Y, U y V de muestras de imagen, donde U y V indican componentes de crominancia (por ejemplo, color) en un espacio de color de luminancia azul y luminancia roja (UV).

40 Se puede emplear SVT para comprimir aún más la información. Específicamente, SVT emplea un bloque 1407 de transformación para comprimir aún más el bloque 1405 residual. El bloque 1407 de transformación contiene una transformación, tal como DCT inversa y/o DST inversa. La diferencia entre el bloque 1403 de predicción y el bloque 1401 de imagen es el ajuste a la transformación que emplea coeficientes de transformación. Al indicar el modo de transformación del bloque 1407 de transformación (por ejemplo, DCT inversa y/o DST inversa) y los coeficientes de transformación correspondientes, el decodificador puede reconstruir el bloque 1405 residual. Cuando no se requiere una reproducción exacta, los coeficientes de transformación se pueden comprimir aún más redondeando ciertos valores para crear un mejor ajuste para la transformación. Este proceso se conoce como cuantificación y se realiza según los parámetros de cuantificación que describen la cuantificación permitida. En consecuencia, los modos de transformación, los coeficientes de transformación y los parámetros de cuantificación del bloque 1407 de transformación se almacenan como información residual transformada en un bloque 1409 residual transformado, que también puede denominarse simplemente bloque residual en algunos casos.

La información de predicción del bloque 1403 de predicción y la información residual transformada del bloque 1409 residual transformado pueden entonces codificarse en un flujo de bits 1411. El flujo de bits 1411 se puede almacenar y/o transmitir a un decodificador. El decodificador puede entonces realizar el proceso a la inversa para recuperar el bloque 1401 de imagen. Específicamente, el decodificador puede emplear la información residual transformada para determinar el bloque 1407 de transformación. El bloque 1407 de transformación puede entonces emplearse junto con el bloque 1409 residual transformado para determinar el bloque 1405 residual. El bloque 1405 residual y el bloque 1403 de predicción se pueden emplear entonces para reconstruir el bloque 1401 de imagen. El bloque 1401 de imagen puede posicionarse entonces con respecto a otros bloques 1401 de imagen decodificados para reconstruir tramas y posicionar tales tramas para recuperar el vídeo codificado.

Se describe ahora SVT con más detalle. Para realizar SVT, el bloque 1407 de transformación se selecciona para que sea menor que el bloque 1405 residual. El bloque 1407 de transformación se emplea para transformar una parte correspondiente del bloque 1405 residual y dejar el resto del bloque residual sin codificación/compresión adicional. Esto se debe a que la información residual generalmente no se distribuye uniformemente en el bloque 1405 residual. SVT emplea un bloque 1407 de transformación menor con una posición adaptativa para capturar la mayoría de la información residual en el bloque 1405 residual, sin requerir que se transforme todo el bloque 1405 residual. Este enfoque puede lograr una mejor eficiencia de codificación que transformar toda la información residual en el bloque 1405 residual. Como el bloque 1407 de transformación es menor que el bloque 1405 residual, SVT emplea un mecanismo para señalar la posición de la transformación con respecto al bloque 1405 residual. Por ejemplo, cuando se aplica SVT para un bloque 1405 residual de tamaño $w \times h$ (por ejemplo, ancho por alto), la información de tamaño y posición del bloque 1407 de transformación puede codificarse en el flujo de bits 1411. Esto permite que el decodificador reconstruya el bloque 1407 de transformación y componga el bloque 1407 de transformación en la posición correcta con respecto al bloque 1409 residual transformado para la reconstrucción del bloque 1405 residual.

Cabe señalar que algunos bloques 1403 de predicción se pueden codificar sin dar como resultado un bloque 1405 residual. Sin embargo, tal caso no da como resultado el uso de SVT y, por lo tanto, no se analiza más. Como se indicó anteriormente, SVT puede emplearse para bloques interpredichos o bloques intrapredichos. Además, SVT puede emplearse en bloques 1405 residuales generados por mecanismos de interpredicción específicos (por ejemplo, compensación de movimiento basada en modelo de traslación), pero no puede emplearse en bloques 1405 residuales generados por otros mecanismos de interpredicción específicos (por ejemplo, modelo afín basado en compensación de movimiento).

La figura 15 ilustra transformaciones 1500 SVT ejemplares que pueden emplearse como un bloque 307 de transformación para codificar un bloque 305 residual. Las transformaciones 1500 SVT se denominan SVT-I, SVT-II y SVT-III. SVT-I se describe como $w_t = w/2$, $h_t = h/2$, donde w_t y h_t indican el ancho y la altura del bloque de transformación, respectivamente, y w y h indican el ancho y la altura del bloque residual, respectivamente. Por ejemplo, el ancho y la altura del bloque de transformación son la mitad del ancho y de la altura del bloque residual. SVT-II se describe como $w_t = w/4$, $h_t = h$ donde las variables son como se describió anteriormente. Por ejemplo, el ancho del bloque de transformación es un cuarto del ancho del bloque residual y la altura del bloque de transformación es igual a la altura del bloque residual. SVT-III se describe como $w_t = w$, $h_t = h/4$ donde las variables son como se describió anteriormente. Por ejemplo, el ancho del bloque de transformación es al ancho del bloque residual y la altura del bloque de transformación es un cuarto de la altura del bloque residual. La información de tipo que indica el tipo de bloque SVT (por ejemplo, SVT-I, SVT-II o SVT-III) se codifica en un flujo de bits para que el decodificador pueda reconstruirlo.

Como puede verse en la figura 15, cada transformación se puede colocar en diversas ubicaciones en relación con el bloque residual. La posición del bloque de transformación está representada por un desplazamiento de posición (x, y) , en la esquina superior izquierda del bloque residual, donde x indica la distancia horizontal entre la esquina superior izquierda del bloque de transformación y la del bloque residual en la unidad de píxeles, e y indica la distancia vertical entre la esquina superior izquierda del bloque de transformación y la del bloque residual en la unidad de píxeles. Cada posición potencial del bloque de transformación dentro del bloque residual se denomina posición candidata. Para un bloque residual, el número de posiciones candidatas es $(w - w_t + 1) \times (h - h_t + 1)$ para un tipo de SVT. Más específicamente, para un bloque residual de 16×16 , cuando se usa SVT-I, hay ochenta y una posiciones candidatas. Cuando se usa SVT-II o SVT-III, hay trece posiciones candidatas. Una vez determinados, los valores x e y del desplazamiento de posición se codifican en un flujo de bits junto con el tipo de bloque SVT empleado. Para reducir la complejidad de SVT-I, se puede seleccionar un subconjunto de treinta y dos posiciones de las ochenta y una posibles posiciones candidatas. Este subconjunto actúa entonces como las posiciones candidatas permitidas para SVT-I.

Un inconveniente de un esquema SVT que emplea transformaciones 1500 SVT es que la codificación de la información de posición SVT como información residual da como resultado una sobrecarga de señalización significativa. Además, la complejidad del codificador puede aumentar significativamente a medida que aumenta el número de posiciones probadas por los procesos de calidad de compresión, tales como la optimización de

la tasa de distorsión (RDO). Dado que el número de posiciones candidatas aumenta con el tamaño del bloque residual, la sobrecarga de señalización puede ser incluso mayor para bloques residuales más grandes, tales como 32×32 o 64×128 . Otro inconveniente de emplear transformaciones 1500 SVT es que el tamaño del bloque de transformación es un cuarto del tamaño del bloque residual. Un bloque de transformación de tal tamaño puede no ser lo suficientemente grande para cubrir la mayor parte de la información residual en el bloque residual en muchos casos.

La figura 16 ilustra transformaciones 1600 SVT ejemplares adicionales que pueden emplearse como un bloque 307 de transformación para codificar un bloque 305 residual. Las transformaciones 1600 SVT se conocen como SVT vertical (SVT-V) y SVT horizontal (SVT-H). Las transformaciones 1600 SVT son similares a las transformaciones 1500 SVT, pero están diseñadas para soportar una sobrecarga de señalización reducida y requisitos de procesamiento menos complejos en el codificador.

SVT-V se describe como $w_t = w/2$ y $h_t = h$, donde las variables son las descritas anteriormente. Por ejemplo, el ancho del bloque de transformación es la mitad del ancho del bloque residual y la altura del bloque de transformación es igual a la altura del bloque residual. SVT-H se describe como $w_t = w$ y $h_t = h/2$, donde las variables son como se describió anteriormente. Por ejemplo, el ancho del bloque de transformación es igual al ancho del bloque residual y la altura del bloque de transformación es la mitad de la altura del bloque residual. SVT-V es similar a SVT-II y SVT-H es similar a SVT-III. En comparación con SVT-II y SVT-III, el bloque de transformación en SVT-V y SVT-H se amplía a la mitad del bloque residual, lo que puede cubrir más información residual en el bloque residual.

Al igual que con las transformaciones 1500 SVT, las transformaciones 1600 SVT pueden incluir varias posiciones candidatas, donde las posiciones candidatas son la posible posición admisible de un bloque de transformación en relación con un bloque residual. Las posiciones candidatas se determinan según el tamaño del escalón de la posición candidata (CPSS). Las posiciones candidatas podrán separarse con un espacio igual especificado por el CPSS. En tal caso, el número de posiciones candidatas se reduce a no más de cinco. El número reducido de posiciones candidatas mitiga la sobrecarga de señalización asociada con la información de posición, ya que la posición seleccionada para una transformación se puede señalar con menos bits. Además, la reducción del número de posiciones candidatas hace que la selección de una posición de transformación sea algorítmicamente más sencilla, lo que permite reducir la complejidad del codificador (por ejemplo, resultando en menos recursos informáticos empleados para codificar).

La figura 17 ilustra posiciones 1700 candidatas de transformación SVT ejemplares en relación con un bloque residual. Específicamente, las posiciones 1700 candidatas de transformación SVT emplean SVT-V y SVT-H (por ejemplo, transformaciones 1600 SVT) colocadas con respecto a un bloque residual. Las posiciones candidatas permitidas para una transformación dependen del CPSS, que además depende de la parte del bloque residual que debe cubrir el bloque de transformación y/o un tamaño del escalón entre las posiciones candidatas. Por ejemplo, el CPSS se puede calcular como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde w y h son el ancho y la altura del bloque residual, respectivamente, y $M1$ y $M2$ son números enteros predeterminados en el intervalo de dos a ocho. Se permiten más posiciones candidatas con un valor $M1$ o $M2$ mayor. Por ejemplo, $M1$ y $M2$ pueden establecerse como ocho. En este caso, un valor del índice de posición (P) que describe la ubicación del bloque de transformación con respecto al bloque residual está entre cero y cuatro.

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde $Th1$ y $Th2$ son números enteros predefinidos que especifican un tamaño del escalón mínimo. $Th1$ y $Th2$ pueden ser números enteros que no son inferiores a dos. En este ejemplo, $Th1$ y $Th2$ se establecen como cuatro, y $M1$ y $M2$ se establecen como ocho. En este ejemplo, diferentes tamaños del bloque pueden tener un número diferente de posiciones candidatas. Por ejemplo, cuando el ancho del bloque residual es ocho, hay dos posiciones candidatas disponibles para SVT-V, específicamente las posiciones candidatas 1700 (a) y (e). Por ejemplo, solo dos posiciones candidatas satisfacen el CPSS cuando el tamaño del escalón, como lo indica $Th1$, es grande y la parte del bloque residual que cubre la transformación, como lo indica $w/M1$, también es grande. Sin embargo, cuando w se establece como dieciséis, la parte del bloque residual que cubre la transformación disminuye debido a un cambio en $w/M1$. Esto da como resultado más posiciones candidatas, en este caso las tres posiciones 1700 (a), (c) y (e) candidatas. Las cinco posiciones candidatas están disponibles cuando el ancho del bloque residual es mayor que dieciséis, mientras que los valores de $Th1$ y $M1$ son como se analizó anteriormente.

También se pueden ver otros ejemplos cuando el CPSS se calcula según otros mecanismos. Específicamente, el CPSS se puede calcular como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H. En este caso, se permiten tres posiciones candidatas para SVT-V (por ejemplo, posiciones 1700 (a), (c) y (e) candidatas) y tres posiciones candidatas para SVT-H (por ejemplo, posiciones 1700 (f), (h) y (j) candidatas) cuando $M1$ y $M2$ se establecen como cuatro. Además, cuando $M1$ y $M2$ se establecen como cuatro, la parte del bloque residual que cubre la transformación aumenta, lo que da como resultado dos posiciones candidatas permitidas de SVT-V (por ejemplo, posiciones 1700 (a) y (e) candidatas) y dos posiciones candidatas permitidas para SVT-H (por

ejemplo, posiciones 1700 (f) y (e) candidatas).

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H como se expuso anteriormente. En este caso, T1 y T2 se establecen como un número entero predefinido, por ejemplo dos, M1 se establece como ocho si $w \geq h$ o se establece como cuatro cuando $w < h$, y M2 se establece como ocho cuando $h \geq w$ o se establece como cuatro cuando $h < w$. Por ejemplo, la parte del bloque residual que cubre la transformación depende de si la altura del bloque residual es mayor que el ancho del bloque residual, o viceversa. En consecuencia, el número de posiciones candidatas para SVT-H o SVT-V depende además de la relación de aspecto del bloque residual.

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H como se expuso anteriormente. En este caso, los valores de M1, M2, Th1 y Th2 se derivan de una estructura de sintaxis de alto nivel en el flujo de bits (por ejemplo, un conjunto de parámetros de secuencia). Por ejemplo, los valores empleados para derivar CPSS se pueden señalar en el flujo de bits. M1 y M2 pueden compartir el mismo valor analizado de un elemento de sintaxis, y Th1 y Th2 pueden compartir el mismo valor analizado de otro elemento de sintaxis.

La figura 18 ilustra posiciones 1800 de transformación SVT ejemplares en relación con un bloque residual. Las posiciones 1800 de transformación SVT se seleccionan de las posiciones candidatas, tales como las posiciones 1700 candidatas de transformación SVT. Específicamente, las posiciones 1800 de transformación SVT seleccionadas pueden codificarse como un índice de posición P. El índice de posición P puede emplearse para determinar un desplazamiento de posición (Z) de la esquina superior izquierda del bloque de transformación con respecto a la esquina superior izquierda del bloque residual. Por ejemplo, esta correlación de posición se puede determinar según $Z = s \times P$, donde s es un CPSS para el bloque de transformación en función del tipo de SVT y calculado como se expuso con respecto a la figura 6. El valor de P puede codificarse como

$\lfloor \frac{w-w_0}{s} \rfloor$ cuando el bloque de transformación es un tipo de SVT-V. El valor de P puede codificarse como

$\lfloor \frac{h-h_0}{s} \rfloor$ cuando el bloque de transformación es un tipo de SVT-H. Más específicamente, (0, 0) puede representar la coordenada de la esquina superior izquierda del bloque residual. En tal caso, la coordenada de la esquina superior izquierda del bloque de transformación es (Z, 0) para SVT-V o (0, Z) para SVT-H.

Como se analiza con más detalle a continuación, el codificador puede codificar el tipo de transformación SVT (por ejemplo, SVT-H o SVT-T) y el tamaño del bloque residual en el flujo de bits empleando indicadores. El decodificador puede entonces determinar el tamaño de la transformación SVT en función del tamaño de la transformación SVT y del tamaño del bloque residual. Una vez que se determina el tamaño de la transformación SVT, el decodificador puede determinar las posiciones candidatas permitidas de la transformación SVT, tales como las posiciones 1700 candidatas de la transformación SVT, según la función CPSS. Dado que el decodificador es capaz de determinar las posiciones candidatas de la transformación SVT, es posible que el codificador no señale las coordenadas del desplazamiento de posición. En su lugar, se puede emplear un código para indicar cuál de las posiciones candidatas se emplea para la transformación correspondiente. Por ejemplo, el índice de posición P puede binarizarse en uno o más contenedores usando código unario truncado para aumentar la compresión. Como ejemplo particular, cuando el valor P está en el intervalo de cero a cuatro, los valores P 0, 4, 2, 3 y 1 se puede binarizar como 0, 01, 001, 0001 y 0000, respectivamente. Este código binario está más comprimido que expresar el valor en base diez del índice de posición. Como otro ejemplo, cuando el valor P está en el intervalo de cero a uno, los valores P 0 y 1 se pueden binarizar como 0 y 1, respectivamente. Como tal, el índice de posición puede aumentar o reducir su tamaño según se desee para señalar una posición del bloque de transformación particular a la luz de las posibles posiciones candidatas para el bloque de transformación.

El índice de posición P puede binarizarse en uno o más contenedores empleando una posición más probable y posiciones restantes menos probables. Por ejemplo, cuando los bloques vecinos izquierdo y superior ya han sido decodificados en el decodificador y, por lo tanto, están disponibles para la predicción, la posición más probable puede establecerse como la posición que cubre la esquina inferior derecha del bloque residual. En un ejemplo, cuando el valor P está en el intervalo de cero a cuatro y la posición cuatro se establece como la posición más probable, los valores P 4, 0, 1, 2 y 3 se binarizan como 1, 000, 001, 010 y 011, respectivamente. Además, cuando el valor P está en el intervalo de cero a dos y la posición dos se establece como la posición más probable, los valores P 2, 0 y 1 se binarizan como 1, 01 y 00, respectivamente. En consecuencia, el índice de posición más probable de las posiciones candidatas se indica con la menor cantidad de bits para reducir la sobrecarga de señalización en los casos más comunes. La probabilidad se puede determinar en función del orden de codificación de los bloques adyacentes reconstruidos. En consecuencia, el decodificador puede inferir el esquema de palabras de código a emplear para un bloque correspondiente en función del esquema de decodificación empleado.

Por ejemplo, en HEVC, el orden de codificación de la unidad de codificación es generalmente de la parte

superior a la parte inferior y de izquierda a derecha. En tal caso, el lado derecho de una unidad de codificación/decodificación actual no está disponible, lo que hace que la esquina superior derecha sea la posición de transformación más probable. Sin embargo, los predictores de vectores de movimiento se derivan de los vecinos espaciales izquierdo y superior. En tal caso, la información residual es estadísticamente más intensa hacia la esquina inferior derecha. En este caso, la posición candidata que cubre la parte inferior derecha es la posición más probable. Además, cuando se utiliza un orden de codificación de unidad de codificación adaptativa, un nodo puede dividirse verticalmente en dos nodos hijos y el nodo hijo derecho puede codificarse antes que el izquierdo. En este caso, el vecino del lado derecho del nodo hijo izquierdo ha sido reconstruido antes que la decodificación/codificación del nodo hijo izquierdo. Además, en tal caso, los píxeles vecinos del lado izquierdo no están disponibles. Cuando el vecino del lado derecho está disponible y el vecino del lado izquierdo no está disponible, es probable que la parte inferior izquierda del bloque residual contenga una gran cantidad de información residual y, por lo tanto, la posición candidata que cubre la parte inferior izquierda del bloque residual se convierta en la posición más probable.

Por lo tanto, el índice de posición P puede binarizarse en uno o más contenedores según si se ha reconstruido el lado derecho junto al bloque residual. En un ejemplo, el valor P está en el intervalo de cero a 2, como lo muestran las posiciones 1800 de transformación SVT. Cuando se ha reconstruido el lado derecho junto al bloque residual, los valores P 0, 2 y 1 se binarizan como 0, 01 y 00. De lo contrario, los valores P 2, 0 y 1 se binarizan como 0, 01 y 00. En otro ejemplo, cuando se ha reconstruido el lado derecho junto al bloque residual, pero no se ha reconstruido el lado izquierdo junto al bloque residual, los valores P 0, 2 y 1 se binarizan como 0, 00 y 01. De lo contrario, los valores P 2, 0 y 1 se binarizan como 0, 00 y 01. En estos ejemplos, la posición correspondiente a un solo contenedor es la posición más probable y las otras dos posiciones son posiciones restantes. Por ejemplo, la posición más probable depende de la disponibilidad del vecino del lado derecho.

La distribución de probabilidad de la mejor posición en el sentido de rendimiento de tasa de distorsión puede ser bastante diferente en los modos de interpredicción. Por ejemplo, cuando el bloque residual corresponde a un bloque de predicción generado por coincidencia de la plantilla con píxeles reconstruidos espacialmente vecinos como la plantilla, lo más probable es que la mejor posición sea dos. Para otros modos de interpredicción, la probabilidad de que la posición dos (o la posición cero cuando el vecino derecho está disponible y el vecino izquierdo no está disponible) sea la mejor posición es menor que la del modo de coincidencia de plantilla. En vista de esto, el modelo de contexto para el primer contenedor del índice de posición P puede decidirse según el modo de interpredicción asociado con el bloque residual. Más específicamente, cuando el bloque residual está asociado con la interpredicción basada en la coincidencia de plantillas, el primer contenedor del índice de posición P usa un primer modelo de contexto. De lo contrario, se usa un segundo modelo de contexto para codificar/decodificar este contenedor.

En otro ejemplo, cuando el bloque residual está asociado con la interpredicción basada en la coincidencia de plantillas, la posición más probable (por ejemplo, la posición dos o la posición cero cuando el vecino derecho está disponible pero el vecino izquierdo no está disponible) se establece directamente como la posición del bloque de transformación y la información de posición no se señala en el flujo de bits. De lo contrario, el índice de posición se señala explícitamente en el flujo de bits.

También cabe señalar que se pueden emplear diferentes transformaciones dependiendo de la posición del bloque de transformación con respecto al bloque residual. Por ejemplo, el lado izquierdo del bloque residual se reconstruye y el lado derecho del bloque residual no se reconstruye, lo que ocurre para la codificación de vídeo con un orden de codificación de unidad de codificación fijo de izquierda a derecha y de la parte superior a la parte inferior (por ejemplo, el orden de codificación en HEVC). En este caso, la posición candidata que cubre la esquina inferior derecha del bloque residual puede emplear DST (por ejemplo, DST tipo siete (DST-7) o DST tipo uno (DST-1) o DST tipo 4 (DST-4)) para una transformación en un bloque de transformación cuando se codifica. En consecuencia, se emplea una transformación DST inversa en el decodificador para la posición candidata correspondiente. Además, una posición candidata que cubre la esquina superior izquierda del bloque residual puede emplear DCT (por ejemplo, DCT tipo ocho (DCT-8) o DCT tipo dos (DCT-2) o DCT tipo cuatro (DCT-4)) para una transformación en el bloque de transformación cuando se codifica. En consecuencia, se emplea una transformación DCT inversa en el decodificador para la posición candidata correspondiente. Esto se debe a que la esquina inferior derecha es la más distante del área reconstruida espacialmente entre las cuatro esquinas en este caso. Además, DST es más efectiva que DCT para transformar la distribución de información residual cuando el bloque de transformación cubre la esquina inferior derecha del bloque residual. Además, DST es más efectiva que DCT para transformar la distribución de información residual cuando el bloque de transformación cubre la esquina superior izquierda del bloque residual. Para el resto de posiciones candidatas, el tipo de transformación puede ser DST o DCT inversa. Por ejemplo, cuando una posición candidata está más cerca de la esquina inferior derecha que de la esquina superior izquierda, se emplea la DST inversa como el tipo de transformación. De lo contrario, se emplea la DCT inversa como el tipo de transformación.

Como ejemplo específico, se pueden permitir tres posiciones candidatas para el bloque de transformación como se muestra en la figura 18. En este caso, la posición cero cubre la esquina superior izquierda y la posición

dos cubre la esquina inferior derecha. La posición uno está en el medio del bloque residual y equidistante de las esquinas izquierda y derecha. Los tipos de transformación se pueden seleccionar como DCT-8, DST-7 y DST-7 para la posición cero, la posición uno y la posición dos, respectivamente, en el codificador. Las transformaciones inversas DCT-8, DST-7 y DST-7 pueden entonces emplearse en el decodificador para la posición cero, la posición uno y la posición dos, respectivamente. En otro ejemplo, los tipos de transformación para la posición cero, la posición uno y la posición dos son DCT-2, DCT-2 y DST-7, respectivamente, en el codificador. Las transformaciones inversas DCT-2, DCT-2 y DST-7 pueden emplearse entonces en el decodificador para la posición cero, la posición uno y la posición dos, respectivamente. Como tal, los tipos de transformación para las posiciones candidatas correspondientes pueden ser predeterminados.

En algunos casos, las transformaciones múltiples dependientes de la posición descritas anteriormente pueden aplicarse solo a bloques de transformación de luminancia. Los bloques de transformación de crominancia correspondientes siempre pueden usar DCT-2 inversa en el proceso de transformación/transformación inversa.

La figura 19 ilustra un ejemplo 1900 de volteo horizontal de muestra residual. En algunos casos, se puede lograr una compresión residual beneficiosa volteando horizontalmente la información residual en el bloque residual (por ejemplo, el bloque 305 residual) antes de aplicar un bloque de transformación (por ejemplo, el bloque 307 de transformación) en el codificador. El ejemplo 1900 ilustra tal volteo horizontal. En este contexto, el volteo horizontal se refiere a rotar las muestras residuales en un bloque residual alrededor de un eje a mitad de camino entre el lado izquierdo del bloque residual y el lado derecho del bloque residual. Tal inversión horizontal ocurre antes de aplicar la transformación (por ejemplo, el bloque de transformación) en el codificador y después de aplicar la transformación inversa (por ejemplo, el bloque de transformación) en el decodificador. Tal volteo puede emplearse cuando ocurren condiciones predefinidas especificadas.

En un ejemplo, el volteo horizontal ocurre cuando el bloque de transformación emplea DST/DST inversa en el proceso de transformación. En este caso, el vecino del lado derecho del bloque residual se codifica/reconstruye antes del bloque actual, y el vecino del lado izquierdo no se codifica/reconstruye antes del bloque actual. El proceso de volteo horizontal intercambia las muestras residuales en la columna i del bloque residual con las muestras residuales en la columna $w-1-i$ del bloque residual. En este contexto, w es el ancho del bloque de transformación e $i=0,1,\dots,(w/2)-1$. El volteo horizontal de las muestras residuales puede aumentar la eficiencia de la codificación haciendo que la distribución residual se ajuste mejor a una transformación DST.

La figura 20 es un diagrama de flujo de un método 2000 ejemplar de decodificación de vídeo con SVT dependiente de la posición que emplea los mecanismos expuestos anteriormente. El método 2000 puede iniciarse en un decodificador al recibir un flujo de bits, tal como el flujo de bits 311. El método 2000 emplea el flujo de bits para determinar los bloques de predicción y los bloques residuales transformados, tal como el bloque 303 de predicción y el bloque 309 residual transformado. El método 2000 también determina bloques de transformación, tal como el bloque 307 de transformación, que se emplea para determinar bloques residuales, tal como el bloque 305 residual. Los bloques residuales y los bloques de predicción se emplean entonces para reconstruir bloques de imagen, tales como el bloque 301 de imagen. Cabe señalar que, aunque el método 2000 se describe desde la perspectiva del decodificador, puede emplearse un método similar (por ejemplo, a la inversa) para codificar el vídeo empleando SVT.

En el bloque 2001, se obtiene un flujo de bits en el decodificador. El flujo de bits se puede recibir desde la memoria o desde un origen de transmisión. El flujo de bits contiene datos que pueden decodificarse en al menos una imagen correspondiente a datos de vídeo de un codificador. Específicamente, el flujo de bits contiene información de partición de bloques que se puede emplear para determinar una unidad de codificación que contiene bloques de predicción y bloques residuales del flujo de bits como se analizó en el mecanismo 300. Como tal, la información de codificación relacionada con las unidades de codificación se puede analizar del flujo de bits y los píxeles de las unidades de codificación se pueden reconstruir en función de la información de codificación, como se analiza a continuación.

En el bloque 2003, un bloque de predicción y un bloque residual transformado correspondiente se obtienen del flujo de bits en función de la información de partición del bloque. Para este ejemplo, el bloque residual transformado ha sido codificado según SVT como se analizó con respecto al mecanismo 300 anterior. El método 2000 reconstruye entonces un bloque residual de tamaño $w \times h$ a partir del bloque residual transformado, como se analiza a continuación.

En el bloque 2005, se determinan el uso de SVT, el tipo de SVT y el tamaño del bloque de transformación. Por ejemplo, el decodificador primero determina si se ha usado SVT en la codificación. Esto se debe a que algunas codificaciones emplean transformaciones que tienen el tamaño del bloque residual. El uso de SVT se puede señalar mediante un elemento de sintaxis en el flujo de bits. Específicamente, cuando a un bloque residual se le permite emplear SVT, se analiza un indicador, tal como `indicador_svt (svt_flag)`, del flujo de bits. Se permite que el bloque residual emplee SVT cuando el bloque residual transformado tiene coeficientes de transformación distintos de cero (por ejemplo, correspondientes a cualquier componente de luminancia o crominancia). Por ejemplo, un bloque residual puede emplear SVT cuando el bloque residual contiene datos

residuales. El indicador SVT indica si el bloque residual se codifica usando un bloque de transformación del mismo tamaño que el bloque residual (por ejemplo, indicador_svt se establece como cero) o el bloque residual se codifica con un bloque de transformación de un tamaño menor que el bloque residual (por ejemplo, indicador_svt se establece como uno). Se puede emplear un indicador de bloque codificado (cbf) para indicar si un bloque residual incluye coeficientes de transformación distintos de cero de un componente de color, como se usa en HEVC. Además, un indicador de bloque codificado raíz (cbf raíz) puede indicar si un bloque residual contiene coeficientes de transformación distintos de cero de cualquier componente de color, como se usa en HEVC. Como ejemplo particular, se permite que un bloque residual use SVT cuando el bloque de imagen se predice usando interpolación y el ancho del bloque residual o la altura del bloque residual cae en un intervalo predeterminado de $[a_1, a_2]$, donde $a_1=16$ y $a_2=64$, $a_1=8$ y $a_2=64$, o $a_1=16$ y $a_2=128$. Los valores de a_1 y a_2 pueden ser valores fijos predeterminados. Los valores también pueden derivarse de un conjunto de parámetros de secuencia (SPS) o una cabecera del segmento en el flujo de bits. Cuando el bloque residual no emplea SVT, el tamaño del bloque de transformación se establece como el ancho y la altura del tamaño del bloque residual. De lo contrario, el tamaño de la transformación se determina en función del tipo de transformación SVT.

Una vez que el decodificador determina que se ha usado SVT para el bloque residual, el decodificador determina el tipo de bloque de transformación SVT usado y deriva el tamaño del bloque de transformación según el tipo de SVT. Los tipos de SVT permitidos para el bloque residual se deciden en función del ancho y la altura del bloque residual. Una transformación SVT-V como se muestra en la figura 16 está permitida si el ancho del bloque residual está en el intervalo $[a_1, a_2]$ tal y como se definieron tales valores anteriormente. Una transformación SVT-H como se muestra en la figura 16 está permitida cuando la altura del bloque residual está en el intervalo $[a_1, a_2]$ tal y como los valores se definieron anteriormente. SVT puede usarse solo para los componentes de luminancia en el bloque residual, o SVT puede usarse para la luminancia y ambos componentes de crominancia en el bloque residual. Cuando SVT solo se usa para el componente de luminancia, la información residual del componente de luminancia es transformado por SVT y los componentes de crominancia son transformados por una transformación del tamaño del bloque residual. Cuando tanto SVT-V como SVT-H están permitidos, un indicador, tal como un indicador_tipo_svt (svt_type_flag), puede codificarse en el flujo de bits. El indicador_tipo_svt indica si SVT-V se usa para el bloque residual (por ejemplo, indicador_tipo_svt se establece como cero) o SVT-H se usa para el bloque residual (por ejemplo, indicador_tipo_svt se establece como uno). Una vez que se determina el tipo de transformación SVT, el tamaño del bloque de transformación se establece según el tipo de SVT señalado (por ejemplo, $w_t=w/2$ y $h_t=h$ para SVT-V, y $w_t=w$ y $h_t=h/2$ para SVT-H). Cuando solo se permite SVT-V o solo se permite SVT-H, el indicador_tipo_svt no se puede codificar en el flujo de bits. En tal caso, el decodificador puede inferir el tamaño del bloque de transformación en función del tipo de SVT permitido.

Una vez que se determinan el tipo y el tamaño de SVT, el decodificador procede al bloque 2007. En el bloque 2007, el decodificador determina la posición de la transformación con respecto al bloque residual así como el tipo de transformación (por ejemplo, DST o DCT). La posición del bloque de transformación se puede determinar según un elemento de sintaxis en el flujo de bits. Por ejemplo, un índice de posición puede señalarse directamente y, por lo tanto, analizarse desde el flujo de bits en algunos ejemplos. En otros ejemplos, la posición se puede inferir como se analiza con respecto a las figuras 5-7. Específicamente, las posiciones candidatas para la transformación se pueden determinar según una función CPSS. La función CPSS puede determinar las posiciones candidatas considerando el ancho del bloque residual, la altura del bloque residual, el tipo de SVT determinado por el bloque 2005, el tamaño del escalón de la transformación, y/o la parte del bloque residual que cubre la transformación. El decodificador puede entonces determinar la posición del bloque de transformación a partir de las posiciones candidatas obteniendo un índice p que incluye un código que señala la posición candidata correcta según la probabilidad de selección de la posición candidata como se analizó con respecto a la figura 18 anterior. Una vez que se conoce la posición del bloque de transformación, el decodificador puede inferir el tipo de transformación empleado por el bloque de transformación como se analizó con respecto a la figura 18 anterior. En consecuencia, el codificador puede seleccionar la transformación inversa correspondiente.

En el bloque 2009, el decodificador analiza los coeficientes de transformación del bloque de transformación en función del tamaño del bloque de transformación determinado en el bloque 2005. Este proceso se puede lograr según los mecanismos de análisis de coeficientes de transformación empleados en HEVC, H.264, y/o codificación de vídeo avanzada (AVC). Los coeficientes de transformación se pueden codificar usando codificación de longitud de ejecución y/o como un conjunto de grupos de coeficientes de transformación (CG). Cabe señalar que el bloque 2009 se puede realizar antes del bloque 2007 en algunos ejemplos.

En el bloque 2011, el bloque residual se reconstruye en función de la posición de transformación, los coeficientes de transformación y el tipo de transformación como se determinó anteriormente. Específicamente, se aplican cuantificación inversa y transformación inversa de tamaño $w_t \times h_t$ a los coeficientes de transformación para recuperar muestras residuales del bloque residual. El tamaño del bloque residual con las muestras residuales es $w_t \times h_t$. La transformación inversa puede ser una DCT inversa o una DST inversa, según el tipo de transformación dependiente de la posición determinada en el bloque 2007. Las muestras

residuales se asignan a una región correspondiente dentro del bloque residual, según la posición del bloque de transformación. Cualquier muestra residual dentro del bloque residual y fuera del bloque de transformación puede establecerse en cero. Por ejemplo, cuando se emplea SVT-V, el número de posiciones candidatas es cinco y el índice de posición indica la quinta posición del bloque de transformación, las muestras residuales reconstruidas se asignan a la región A en las posiciones 1700 (e) candidatas de transformación y la región de tamaño $(w/2) \times h$ a la izquierda de la región A tiene cero muestras residuales.

En el bloque 2013 opcional, la información de bloque residual del bloque reconstruido se puede voltear horizontalmente como se expuso con respecto a la figura 19. Como se indicó anteriormente, esto puede ocurrir cuando el bloque de transformación en el decodificador emplea DST inversa, el bloque vecino del lado derecho ya está reconstruido y el vecino del lado izquierdo aún no está reconstruido. Específicamente, el codificador puede voltear horizontalmente el bloque residual antes de aplicar una transformación DST en el caso mencionado anteriormente para aumentar la eficiencia de codificación. En consecuencia, puede emplearse el bloque 2013 opcional para corregir tal volteo horizontal en el codificador para crear un bloque reconstruido preciso.

En el bloque 2015, el bloque residual reconstruido se puede componer con el bloque de predicción para generar el bloque de imagen reconstruido que contiene muestras como parte de una unidad de codificación. También se puede aplicar un proceso de filtrado en las muestras reconstruidas, tal como el filtro de desbloqueo y el procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC. El bloque de imagen reconstruido puede entonces combinarse con otros bloques de imagen decodificados de manera similar para generar tramas de un archivo de multimedia/de vídeo. El archivo de multimedia reconstruido puede entonces mostrarse a un usuario en un monitor u otro dispositivo de visualización.

Cabe señalar que se puede emplear una implementación equivalente del método 2000 para generar las muestras reconstruidas en el bloque residual. Específicamente, las muestras residuales del bloque de transformación se pueden componer directamente con el bloque de predicción en la posición indicada por la información de posición del bloque de transformación, sin recuperar primero el bloque residual.

En resumen, la descripción anterior incluye mecanismos que emplean de forma adaptativa múltiples tipos de transformación para el bloque de transformación en diferentes posiciones. Además, la descripción permite voltear horizontalmente las muestras residuales en el bloque residual para favorecer la eficiencia de la codificación. Esto ocurre cuando el bloque de transformación usa DST y DST inversa en el codificador y decodificador, respectivamente, y cuando el bloque vecino derecho está disponible y el vecino izquierdo no está disponible. Además, la descripción incluye mecanismos para soportar la codificación de la información de posición en el flujo de bits en función del modo de interpredicción asociado con el bloque residual.

Se hace referencia al siguiente documento, que proporciona detalles adicionales sobre SVT: C. Zhang, K. Ugur, J. Lainema, A. Hallapuro and M. Gabbouj, "Video Coding Using Spatially Varying Transform", IEEE Trans. Proceso de imagen., vol. 21, n.º 2, febrero de 2011, págs. 127-140.

La presente descripción introduce además otro esquema de SVT mejorado. La mejora es que el tipo de transformación horizontal y el tipo de transformación vertical de un bloque SVT se determinan en función del tipo de SVT y de la posición del bloque SVT. La transformación horizontal puede ser diferente de la transformación vertical. Anteriormente, el tipo de transformación horizontal y el tipo de transformación vertical de un bloque SVT se determinaban únicamente en función de la posición del bloque SVT y siempre son iguales.

Una primera implementación describe el proceso de decodificación de un bloque residual. Se decodifica un flujo de bits que contiene al menos una imagen de datos de vídeo. Una imagen se divide en una pluralidad de regiones de imagen rectangulares y cada región corresponde a una unidad de árbol de codificación (CTU). Una CTU se divide en una pluralidad de bloques, tales como las unidades de codificación en HEVC, según la información de partición de bloque contenida en el flujo de bits. La información de codificación de los bloques se analiza del flujo de bits y los píxeles de los bloques se reconstruyen en función de la información de codificación.

En esta implementación, SVT está restringida a ser usada para bloques interpredichos. SVT también se puede usar para bloques intrapredichos. En un método más avanzado, SVT puede permitirse para bloques que usan un método de interpredicción específico (por ejemplo, compensación de movimiento basada en modelo de traslación) pero no permitirse para bloques que usan algún otro método de interpredicción (por ejemplo, compensación de movimiento basada en modelo afín). En otro ejemplo, SVT puede permitirse para bloques de predicción que usan el modo de combinación o modo AMVP (predicción avanzada del vector de movimiento) con precisión de diferencia de vector de movimiento de 1/4-pel, pero no se permite para bloques de predicción que usan el modo de combinación afín, el modo interafín o el modo AMVP con precisión de diferencia de vector de movimiento de 1-pel o 4-pel. En otro ejemplo, SVT puede permitirse para los bloques de predicción que usan el modo de combinación con un índice de combinación menor que 2, pero no se permite para bloques de predicción que usan el modo de combinación con un índice de combinación no menor que 2. En el estándar

H.265/HEVC se puede hacer referencia al modo de combinación y al modo AMVP. En el códec JEM (modelo de escaneo conjunto) de JVET (equipo de escaneo de vídeo conjunto) se puede hacer referencia al modo de combinación afín y al modo interafín.

- 5 En un ejemplo, un bloque puede referirse a una unidad de codificación, donde la unidad de codificación puede contener un bloque de predicción y un bloque residual. El bloque de predicción puede contener todas las muestras de predicción de la unidad de codificación, el bloque residual puede contener todas las muestras residuales de la unidad de codificación y el bloque de predicción tiene el mismo tamaño que el bloque residual.
- 10 En otro ejemplo, un bloque puede referirse a una unidad de codificación, una unidad de codificación puede contener dos bloques de predicción y un bloque residual, cada bloque de predicción puede contener una parte de las muestras de predicción de la unidad de codificación y un bloque residual puede contener todas las muestras residuales de la unidad de codificación. En otro ejemplo, un bloque puede referirse a una unidad de codificación, una unidad de codificación puede contener dos bloques de predicción y cuatro bloques residuales.
- 15 El patrón de partición de bloques residuales en una unidad de codificación puede señalarse en el flujo de bits, tal como un árbol cuádruple residual (RQT) en HEVC.

Un bloque puede contener solo el componente Y de muestras de imagen (o píxeles), o puede contener los componentes Y, U y V de muestras de imagen.

- 20 Un bloque residual R_0 de tamaño $w \times h$ puede reconstruirse mediante las siguientes etapas.

Etapa 1. Determinar el tamaño del bloque de transformación del bloque residual R_0 .

- 25 Etapa 1.1. Determinar el uso de SVT según un elemento de sintaxis. Para un bloque residual al que se le permite usar SVT, si el bloque residual tiene coeficientes de transformación distintos de cero del componente Y (o tiene coeficientes de transformación distintos de cero de cualquier componente de color), un indicador (es decir, `indicador_svt`) se analiza a partir el flujo de bits. El indicador indica si el bloque residual se codifica usando un bloque de transformación del mismo tamaño que el bloque residual (por ejemplo, `indicador_svt = 0`) o el bloque residual se codifica con un bloque de transformación de un tamaño menor que el bloque residual (por ejemplo, `indicador_svt = 1`). Puede indicarse si un bloque tiene coeficientes de transformación distintos de cero de un componente de color mediante un indicador de bloque codificado (cbf) del componente de color, como se usa en HEVC. Puede indicarse si un bloque tiene coeficientes de transformación distintos de cero de cualquier componente de color mediante un indicador de bloque codificado raíz (cbf raíz), como se usa en HEVC.
- 30
- 35

En un ejemplo, un bloque puede usar SVT si se cumplen las siguientes condiciones:

- 1) el bloque se predice usando interpredicción;
- 40 2) el ancho del bloque o la altura del bloque cae en un intervalo predeterminado $[a_1, a_2]$, por ejemplo, $a_1=16$ y $a_2=64$, o $a_1=8$ y $a_2=64$, o $a_1=16$ y $a_2=128$. El valor de a_1 y a_2 pueden ser valores fijos. El valor también se puede derivar de un conjunto de parámetros de secuencia (SPS) o una cabecera del segmento.

En un ejemplo, un bloque puede usar SVT si se cumplen las siguientes condiciones:

- 45 1) el bloque se predice usando el modo de combinación con un índice de combinación menor que un umbral (por ejemplo, 1 o 2 o 3) o usando el modo AMVP con precisión de diferencia de vector de movimiento de 1/4-pel;
- 50 2) una dimensión del bloque cae en un intervalo predeterminado $[a_1, a_2]$, y la otra dimensión del bloque no es mayor que un umbral a_3 , por ejemplo, $a_1=8$, $a_2=32$ y $a_3=32$. El parámetro a_1 puede establecerse como el doble del tamaño de transformación mínimo, a_2 y a_3 pueden establecerse como el tamaño de transformación máximo. El valor de a_1 , a_2 y a_3 pueden ser valores fijos. El valor también se puede derivar de un conjunto de parámetros de secuencia (SPS) o una cabecera del segmento.
- 55

Si el bloque no usa SVT, el tamaño del bloque de transformación se establece como $w \times h$. De lo contrario, se aplica la Etapa 1.2 para decidir el tamaño de la transformación.

- 60 Etapa 1.2. Determinar el tipo de SVT según un elemento de sintaxis y obtener el tamaño del bloque de transformación según el tipo de SVT. Los tipos de SVT permitidos para el bloque residual se deciden en función del ancho y la altura del bloque residual. SVT-V está permitido si w está en el intervalo $[a_1, a_2]$ y h no es mayor que a_3 ; SVT-H está permitido si h está en el intervalo $[a_1, a_2]$ y w no es mayor que a_3 . SVT puede usarse solo para el componente Y, o puede usarse para los tres componentes. Cuando SVT solo se usa para el componente Y, el residual del componente Y se transforma mediante SVT y los componentes U y V se transforman según el tamaño del bloque residual.
- 65

Cuando se permiten tanto SVT-V como SVT-H, se analiza un indicador (a saber, indicador_tipo_svt) del flujo de bits, lo que indica si se usa SVT-V (por ejemplo, indicador_tipo_svt = 0) o se usa SVT-H (por ejemplo, indicador_tipo_svt = 1) para el bloque residual, y el tamaño del bloque de transformación se establece según el tipo de SVT señalado (es decir, $w_t=w/2$ y $h_t=h$ para SVT-V, y $w_t=w$ y $h_t=h/2$ para SVT-H). Cuando solo se permite SVT-V o solo se permite SVT-H, el indicador_tipo_svt no se analiza del flujo de bits y el tamaño del bloque de transformación se establece según el tipo de SVT permitido.

Etapa 2. Determinar la posición del bloque de transformación según un elemento de sintaxis y determinar el tipo de transformación para el bloque de transformación en función del tipo de SVT y la información de posición del bloque de transformación.

Etapa 2.1: Determinar la posición del bloque de transformación según un elemento de sintaxis.

Se analiza un índice de posición P del flujo de bits, y el desplazamiento de posición Z de la esquina superior izquierda del bloque de transformación a la esquina superior izquierda del bloque residual se determina como $Z = s \times P$, donde s es un tamaño del escalón de posición candidato (CPSS). El valor de P está entre

0, 1, ..., $\frac{w_t - w_t - t}{s}$ si se usa SVT-V, o el valor de P está entre 0, 1, ..., $\frac{h_t - h_t - t}{s}$ si se usa SVT-H. Más específicamente, si (0, 0) representa la coordenada de la esquina superior izquierda del bloque residual, la coordenada de la esquina superior izquierda del bloque de transformación es (Z, 0) para SVT-V o (0, Z) para SVT-H.

En un ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde w y h son el ancho y la altura del bloque residual, respectivamente, y M1 y M2 son números enteros predeterminados en el intervalo de 2 a 8. Se permiten más posiciones candidatas con un valor mayor de M1 o M2. En este ejemplo, tanto M1 como M2 se establecen como 8. Por lo tanto, el valor de P está entre 0 y 4. Las posiciones candidatas se ilustran en la figura 21.

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde Th1 y Th2 son números enteros predefinidos que especifican un tamaño del escalón mínimo. Th1 y Th2 son números enteros no menores que 2. En este ejemplo, Th1 y Th2 se establecen como 4, y M1 y M2 se establecen como 8. En este ejemplo, los diferentes tamaños del bloque pueden tener un número diferente de posiciones candidatas. Por ejemplo, cuando $w=8$, están disponibles 2 posiciones candidatas para elegir (como se ilustra en la figura 21 (a) y la figura 21 (e)); cuando $w=16$, están disponibles 3 posiciones candidatas para elegir (como se ilustra en la figura 21(a), la figura 21(c) y la figura 21(e)); cuando $w>16$, hay 5 posiciones disponibles para elegir.

En otro ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde M1 y M2 se establecen como 4. En consecuencia, se permiten 3 posiciones candidatas.

En otro ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde M1 y M2 se establecen como 2. En consecuencia, se permiten 2 posiciones candidatas.

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde T1 y T2 se establecen como 2, M1 se establece como 8 si $w \geq h$ o se establece como 4 si $w < h$, y M2 se establece como 8 si $h \geq w$ o se establece como 4 si $h < w$. En este caso, el número de posiciones candidatas para SVT-H o SVT-V depende además de la relación de aspecto del bloque residual.

En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde los valores de M1, M2, Th1 y Th2 se derivan de una estructura de sintaxis de alto nivel en el flujo de bits (por ejemplo, el conjunto de parámetros de secuencia). M1 y M2 pueden compartir el mismo valor analizado de un elemento de sintaxis, y Th1 y Th2 pueden compartir el mismo valor analizado de otro elemento de sintaxis.

El índice de posición P puede binarizarse en uno o más contenedores usando código unario truncado. Por ejemplo, si el valor P está en el intervalo de 0 a 4, los valores P 0, 4, 2, 3 y 1 se binarizan como 0, 01, 001, 0001 y 0000, respectivamente; si el valor P está en el intervalo de 0 a 1, los valores P 0 y 1 se binarizan como 0 y 1, respectivamente.

El índice de posición P puede binarizarse en uno o más contenedores usando una posición más probable y varias posiciones restantes. Cuando los vecinos izquierdo y superior están disponibles, la posición más probable se puede establecer como la posición que cubre la esquina inferior derecha del bloque residual. En un ejemplo, si el valor P está en el intervalo de 0 a 4 y la posición 4 se establece como la posición más probable, los valores P 4, 0, 1, 2 y 3 se binarizan como 1, 000, 001, 010 y 011, respectivamente; si el valor P está en el intervalo de 0 a 2 y la posición 2 se establece como la posición más probable, los valores P 2, 0 y 1 se binarizan

como 1, 01 y 00, respectivamente.

5 Etapa 2.2: determinar el tipo de transformación para el bloque de transformación en función del tipo de SVT y la información de posición del bloque de transformación. El tipo de transformación incluye la transformación horizontal y la transformación vertical de una transformación separable 2D.

10 Tomando como ejemplo el caso de que se permitan 3 posiciones candidatas, como se muestra en la figura 22. La posición 0 cubre la esquina superior izquierda y la posición 2 cubre la esquina inferior derecha. La posición 1 está en el medio del bloque residual. Hay tres posiciones para SVT-V y SVT-H, como se muestra en la figura 22.

15 Una transformación bidimensional puede separarse en una transformación horizontal y una transformación vertical unidimensionales. Se puede realizar una transformación 2D directa que convierte el residual en coeficientes de transformación aplicando primero una transformación horizontal en un bloque residual para generar un bloque TA y luego aplicando una transformación vertical en el bloque TA para generar un bloque de coeficientes de transformación, como se implementa en el códec JEM. En consecuencia, se puede realizar una transformación 2D inversa que convierte los coeficientes de transformación de vuelta al residual aplicando primero una transformación vertical inversa en un bloque de coeficientes de transformación para generar un bloque TB y luego aplicando una transformación inversa horizontal en el bloque TB para generar un bloque residual, como se implementa en el códec JEM.

25 En un ejemplo, las transformaciones horizontal y vertical para la posición 0 de SVT-V son DCT-8 y DST-7; las transformaciones horizontal y vertical para la posición 1 de SVT-V son DST-1 y DST-7; las transformaciones horizontal y vertical para la posición 2 de SVT-V son DST-7 y DST-7; las transformaciones horizontal y vertical para la posición 0 de SVT-H son DST-7 y DCT-8; las transformaciones horizontal y vertical para la posición 1 de SVT-H son DST-7 y DST-1; las transformaciones horizontal y vertical para la posición 2 de SVT-H son DST-7 y DST-7, como se indica en la Tabla I. En este ejemplo, la transformación vertical para SVT-V y la transformación horizontal para SVT-H se establecen como DST-7, y la otra transformación se basa en la posición de SVT.

30 Tabla I. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-8	DST-7
SVT-V, posición 1	DST-1	DST-7
SVT-V, posición 2	DST-7	DST-7
SVT-H, posición 0	DST-7	DCT-8
SVT-H, posición 1	DST-7	DST-1
SVT-H, posición 2	DST-7	DST-7

35 En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. En consecuencia, en un ejemplo, las transformaciones horizontal y vertical para la posición 0 de SVT-V son DCT-4 y DST-4; las transformaciones horizontal y vertical para la posición 1 de SVT-V son DST-1 y DST-4; las transformaciones horizontal y vertical para la posición 2 de SVT-V son DST-4 y DST-4; las transformaciones horizontal y vertical para la posición 0 de SVT-H son DST-4 y DCT-4; las transformaciones horizontal y vertical para la posición 1 de SVT-H son DST-4 y DST-1; las transformaciones horizontal y vertical para la posición 2 de SVT-H son DST-4 y DST-4, como se indica en la Tabla I'. En este ejemplo, la transformación vertical para SVT-V y la transformación horizontal para SVT-H se establecen como DST-4, y la otra transformación se basa en la posición de SVT.

45 Tabla I'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-4	DST-4
SVT-V, posición 1	DST-1	DST-4
SVT-V, posición 2	DST-4	DST-4
SVT-H, posición 0	DST-4	DCT-4
SVT-H, posición 1	DST-4	DST-1
SVT-H, posición 2	DST-4	DST-4

En otro ejemplo, la transformación horizontal y la transformación vertical para diferentes tipos y posiciones de SVT se indica en la Tabla II. En este ejemplo, la transformación vertical para SVT-V y la transformación horizontal para SVT-H se establecen como DCT-2, y la otra transformación se basa en la posición de SVT.

5

Tabla II. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-8	DCT-2
SVT-V, posición 1	DST-1	DCT-2
SVT-V, posición 2	DST-7	DCT-2
SVT-H, posición 0	DCT-2	DCT-8
SVT-H, posición 1	DCT-2	DST-1
SVT-H, posición 2	DCT-2	DST-7

10 En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. Por consiguiente,

Tabla II'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-4	DCT-2
SVT-V, posición 1	DST-1	DCT-2
SVT-V, posición 2	DST-4	DCT-2
SVT-H, posición 0	DCT-2	DCT-4
SVT-H, posición 1	DCT-2	DST-1
SVT-H, posición 2	DCT-2	DST-4

15 En otro ejemplo, la transformación horizontal y la transformación vertical para diferentes tipos y posiciones de SVT se indican en la Tabla III. En este ejemplo, la transformación horizontal y la transformación vertical se deciden solo por la posición de SVT.

20 Tabla III. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-8	DCT-8
SVT-V, posición 1	DST-1	DST-1
SVT-V, posición 2	DST-7	DST-7
SVT-H, posición 0	DCT-8	DCT-8
SVT-H, posición 1	DST-1	DST-1
SVT-H, posición 2	DST-7	DST-7

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. Por consiguiente,

25 Tabla III'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-4	DCT-4
SVT-V, posición 1	DST-1	DST-1
SVT-V, posición 2	DST-4	DST-4
SVT-H, posición 0	DCT-4	DCT-4
SVT-H, posición 1	DST-1	DST-1

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-H, posición 2	DST-4	DST-4

En otro ejemplo, la transformación horizontal y la transformación vertical para diferentes tipos y posiciones de SVT se indican en la Tabla IV.

- 5 Tabla IV. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DST-7	DCT-8
SVT-V, posición 1	DST-7	DST-1
SVT-V, posición 2	DST-7	DST-7
SVT-H, posición 0	DCT-8	DST-7
SVT-H, posición 1	DST-1	DST-7
SVT-H, posición 2	DST-7	DST-7

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. Por consiguiente,

10

Tabla IV'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DST-4	DCT-4
SVT-V, posición 1	DST-4	DST-1
SVT-V, posición 2	DST-4	DST-4
SVT-H, posición 0	DCT-4	DST-4
SVT-H, posición 1	DST-1	DST-4
SVT-H, posición 2	DST-4	DST-4

En otro ejemplo, la transformación horizontal y la transformación vertical para diferentes tipos y posiciones de SVT se indican en la Tabla V.

15

Tabla V. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-8	DST-7
SVT-V, posición 1	DCT-2	DST-7
SVT-V, posición 2	DST-7	DST-7
SVT-H, posición 0	DST-7	DCT-8
SVT-H, posición 1	DST-7	DCT-2
SVT-H, posición 2	DST-7	DST-7

- 20 En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. Por consiguiente,

Tabla V'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-4	DST-4
SVT-V, posición 1	DCT-2	DST-4
SVT-V, posición 2	DST-4	DST-4
SVT-H, posición 0	DST-4	DCT-4

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-H, posición 1	DST-4	DCT-2
SVT-H, posición 2	DST-4	DST-4

En otro ejemplo, la transformación horizontal y la transformación vertical para diferentes tipos y posiciones de SVT se indican en la Tabla VI.

- 5 Tabla VI. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-8	DCT-2
SVT-V, posición 1	DCT-2	DCT-2
SVT-V, posición 2	DST-7	DCT-2
SVT-H, posición 0	DCT-2	DCT-8
SVT-H, posición 1	DCT-2	DCT-2
SVT-H, posición 2	DCT-2	DST-7

En una forma de implementación alternativa, DST-7 puede reemplazarse con DST-4, y DCT-8 puede reemplazarse con DCT-4 a lo largo de la presente descripción. Por consiguiente,

10

- Tabla VI'. Un ejemplo de transformaciones horizontales y verticales 1-D para diferentes tipos y posiciones de SVT.

tipo de SVT, posición	transformación horizontal	transformación vertical
SVT-V, posición 0	DCT-4	DCT-2
SVT-V, posición 1	DCT-2	DCT-2
SVT-V, posición 2	DST-4	DCT-2
SVT-H, posición 0	DCT-2	DCT-4
SVT-H, posición 1	DCT-2	DCT-2
SVT-H, posición 2	DCT-2	DST-4

15

Las transformaciones múltiples dependientes de la posición se pueden aplicar solo al bloque de transformación de luminancia, y los bloques de transformación de crominancia correspondientes siempre usan la DCT-2 inversa en el proceso de transformación inversa.

20

Etapa 3. Analizar los coeficientes de transformación del bloque de transformación en función del tamaño del bloque de transformación.

Este es un proceso comúnmente usado en la decodificación de vídeo, tal como el análisis de coeficientes de transformación en HEVC o H.264/AVC. Los coeficientes de transformación pueden codificarse usando codificación de longitud de ejecución o codificarse de forma más sofisticada como un conjunto de grupos de coeficientes de transformación (CG).

25

La Etapa 3 puede llevarse a cabo antes de la Etapa 2.

30

Etapa 4. Reconstruir el bloque residual R_0 en función de los coeficientes de transformación y la posición del bloque de transformación y el tipo de transformación inversa.

35

Se aplican cuantificación inversa y transformación inversa de tamaño $w_t \times h_t$ a los coeficientes de transformación para recuperar muestras residuales. El tamaño de las muestras residuales es $w_t \times h_t$, que es igual que el tamaño del bloque de transformación. La transformación inversa es una transformación separable 2-D. El bloque de coeficiente de transformación decuantificado se transforma primero mediante una transformación vertical inversa para generar un bloque TC y luego el bloque TC se transforma mediante una transformación horizontal inversa, en la que la transformación horizontal inversa y la transformación vertical inversa se deciden en la Etapa 2.2 en función de la posición del bloque de transformación o en función de tanto la posición del bloque de transformación como del tipo de SVT del bloque de transformación.

40

Las muestras residuales se asignan a una región correspondiente dentro del bloque residual R_0 , según la posición del bloque de transformación, y las muestras restantes dentro del bloque residual se establecen como

ceros. Por ejemplo, si se usa SVT-V y el número de posiciones candidatas es 5 y el índice de posición es 4, las muestras residuales reconstruidas se asignan a la región A en la figura 21(e) y la región de tamaño $(w/2) \times h$ a la izquierda de la región A tiene residual cero.

- 5 Después de realizar las Etapas 1 a 4, el bloque residual reconstruido se puede componer con un bloque de predicción para generar las muestras reconstruidas en una unidad de codificación. Posteriormente, se puede aplicar un proceso de filtrado a las muestras reconstruidas, tal como el filtro de desbloqueo y el procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC.
- 10 A diferencia de las soluciones existentes, la presente solución usa de forma adaptativa múltiples tipos de transformación para el bloque de transformación en función del tipo de SVT y la información de posición.

La presente descripción introduce un esquema SVT mejorado. Se usan dos tipos de bloques SVT para la codificación residual, denominados SVT-H y SVT-V, como se muestra en la figura 23. Estos son (1) SVT-V:

- 15 $w_t = w/2$ y $h_t = h$; y (2) SVT-H: $w_t = w$ y $h_t = h/2$.

SVT-V es similar a SVT-II y SVT-H es similar a SVT-III. En comparación con SVT-II y SVT-III, el bloque de transformación en SVT-V y SVT-H se amplía a la mitad del bloque residual, lo que puede cubrir más residual en el bloque residual. Tenga en cuenta que SVT-II o SVT-H pueden usarse para reemplazar SVT-V o SVT-H.

- 20 Las posiciones candidatas se deciden mediante un tamaño del escalón de la posición candidata (CPSS). En consecuencia, las posiciones candidatas se separan con un espacio igual especificado por el CPSS. El número de posiciones candidatas se reduce a no más de 5 en diferentes implementaciones, lo que mitiga la sobrecarga de la información de posición, así como la complejidad del codificador, para decidir la mejor posición del bloque de transformación.

El CPSS indica la distancia entre dos posiciones candidatas adyacentes (por ejemplo, 1 en [1]), y se calcula en función del ancho o de la altura del bloque residual. Más específicamente, el tamaño del escalón de la posición candidata para SVT-V se determina como $h_b = \max(w/M1, Th1)$, donde M1 y Th1 son números enteros predefinidos, y $\max(a,b)$ es la operación de selección del valor máximo entre a y b. De manera similar, los tamaños del escalón de la posición candidata para SVT-H se determinan como $w_b = \max(w/M2, Th2)$, donde M2 y Th2 son números enteros predefinidos. M1 y M2 son números enteros entre 2 y 8, por ejemplo, M1=M2=8, o M1=M2=4, o M1=M2=2. Th1 y Th2 son números enteros no menores que 2. M1, M2, Th1 y Th2 se pueden establecer como valores fijos o se pueden derivar de una estructura de sintaxis de alto nivel en el flujo de bits (por ejemplo, el conjunto de parámetros de secuencia).

- 30 La codificación de la información de SVT (por ejemplo, el uso de SVT para un bloque residual, el tipo de SVT y la posición del bloque de transformación) es diferente de la de [1], y se mejora aun más analizando los coeficientes de transformación del bloque residual.

Para el caso donde SVT-V: $w_t = w/2$ y $h_t = h$, la información de SVT puede derivarse parcialmente de los coeficientes de transformación decodificados. En un ejemplo, la información de posición puede derivarse de la suma de los niveles de los coeficientes de transformación. En otro ejemplo, la información de tipo de SVT se deriva de la suma de un primer grupo de coeficientes de transformación (por ejemplo, los primeros 16 coeficientes en un orden de escaneo), y la información de posición se deriva de la suma de un segundo grupo de coeficientes de transformación (por ejemplo, el resto de coeficientes en el bloque de transformación).

- 40 Para el caso donde SVT-H: $w_t = w$ y $h_t = h/2$, el número de posiciones candidatas se determina según la información del coeficiente de transformación, y la información de posición se decodifica en función del número de posiciones candidatas.

Implementación 1

Esta implementación describe el proceso de decodificación de un bloque residual. Se decodifica un flujo de bits que contiene al menos una imagen de datos de vídeo. Una imagen se divide en una pluralidad de regiones de imagen rectangulares y cada región corresponde a una unidad de árbol de codificación (CTU). Una CTU se divide en una pluralidad de bloques, tal como las unidades de codificación en codificación de vídeo de alta eficiencia (HEVC) según la información de partición del bloque contenido en el flujo de datos. La información de codificación de los bloques se analiza del flujo de bits y los píxeles de los bloques se reconstruyen en función de la información de codificación.

En esta implementación, el uso de SVT está restringido para bloques interpredichos. SVT también se puede usar para bloques intrapredichos. En un método más avanzado, SVT puede permitirse para bloques que usan un método de interpredicción específico (por ejemplo, compensación de movimiento basada en modelo de traslación) pero no permitirse para bloques que usan algún otro método de interpredicción (por ejemplo, compensación de movimiento basada en modelo afín).

- 65

En un ejemplo, un bloque puede referirse a una unidad de codificación, donde la unidad de codificación puede contener un bloque de predicción y un bloque residual. El bloque de predicción puede contener todas las muestras de predicción de la unidad de codificación, el bloque residual puede contener todas las muestras residuales de la unidad de codificación y el bloque de predicción tiene el mismo tamaño que el bloque residual.
 En otro ejemplo, un bloque puede referirse a una unidad de codificación, una unidad de codificación puede contener dos bloques de predicción y un bloque residual, cada bloque de predicción puede contener una parte de las muestras de predicción de la unidad de codificación y un bloque residual puede contener todas las muestras residuales de la unidad de codificación. En otro ejemplo, un bloque puede referirse a una unidad de codificación, una unidad de codificación puede contener dos bloques de predicción y cuatro bloques residuales. El patrón de partición de bloques residuales en una unidad de codificación puede señalarse en el flujo de bits, tal como un árbol cuádruple residual (RQT) en HEVC.

Un bloque puede contener solo el componente Y de muestras de imagen (o píxeles), o puede contener los componentes Y, U y V de muestras de imagen.

Un bloque residual R_0 de tamaño $w \times h$ puede reconstruirse mediante las siguientes etapas.

Etapa 1. Determinar el tamaño del bloque de transformación del bloque residual R_0 .

Etapa 1.1. Determinar el uso de SVT según un elemento de sintaxis. Para un bloque residual al que se le permite usar SVT, si el bloque residual tiene coeficientes de transformación distintos de cero del componente Y (o tiene coeficientes de transformación distintos de cero de cualquier componente de color), un indicador (es decir, `indicador_svt`) se analiza del flujo de bits, lo que indica si el bloque residual está codificado usando un bloque de transformación del mismo tamaño que el bloque residual (por ejemplo, `indicador_svt = 0`) o el bloque residual está codificado con un bloque de transformación de un tamaño menor que el bloque residual (por ejemplo, `indicador_svt = 1`). Si un bloque tiene coeficientes de transformación distintos de cero de un componente de color, puede indicarse mediante un indicador de bloque codificado (`cbf`) del componente de color, como se usa en HEVC. Puede indicarse si un bloque tiene coeficientes de transformación distintos de cero de cualquier componente de color mediante un indicador de bloque codificado raíz (`cbf raíz`), como se usa en HEVC.

En un ejemplo, un bloque puede usar SVT si se cumplen las siguientes condiciones:

- 1) el bloque se predice usando interpredicción;
 - 2) el ancho del bloque o la altura del bloque cae en un intervalo predeterminado $[a1, a2]$, por ejemplo, $a1=16$ y $a2=64$, o $a1=8$ y $a2=64$, o $a1=16$ y $a2=128$. El valor de $a1$ y $a2$ pueden ser valores fijos. El valor también se puede derivar de un conjunto de parámetros de secuencia (SPS) o una cabecera del segmento.
- Si el bloque no usa SVT, el tamaño del bloque de transformación se establece como $w \times h$. De lo contrario, se aplica la Etapa 1.2 para decidir el tamaño de la transformación.

Etapa 1.2. Determinar el tipo de SVT según un elemento de sintaxis y obtener el tamaño del bloque de transformación según el tipo de SVT. Los tipos de SVT permitidos para el bloque residual se deciden en función del ancho y la altura del bloque residual. SVT-V está permitido si w está en el intervalo $[a1, a2]$; SVT-H está permitido si h está en el intervalo $[a1, a2]$. SVT puede usarse solo para el componente Y, o puede usarse para todos los tres componentes. Cuando SVT solo se usa para el componente Y, el residual del componente Y se transforma mediante SVT y los componentes U y V se transforman según el tamaño del bloque residual.

Cuando se permiten tanto SVT-V como SVT-H, se analiza un indicador (a saber, `indicador_tipo_svt`) del flujo de bits, lo que indica si se usa SVT-V (por ejemplo, `indicador_tipo_svt = 0`) o se usa SVT-H (por ejemplo, `indicador_tipo_svt = 1`) para el bloque residual, y el tamaño del bloque de transformación se establece según el tipo de SVT señalado (es decir, $w_t=w/2$ y $h_t=h$ para SVT-V, y $w_t=w$ y $h_t=h/2$ para SVT-H). Cuando solo se permite SVT-V o solo se permite SVT-H, el `indicador_tipo_svt` no se analiza del flujo de bits y el tamaño del bloque de transformación se establece según el tipo de SVT permitido.

Etapa 2. Determinar la posición del bloque de transformación según un elemento de sintaxis.

Se analiza un índice de posición P del flujo de bits, y se determina el desplazamiento de posición Z de la esquina superior izquierda del bloque de transformación a la esquina superior izquierda del bloque residual

como $Z = s \times P$, donde s es un CPSS. El valor de P está entre $0, 1, \dots, \frac{w-h-1}{s}$ si se usa SVT-V, o el valor

$$s = 1, \dots, \frac{w-h-t}{s}$$

de P está entre s si se usa SVT-H. Más específicamente, si (0, 0) representa la coordenada de la esquina superior izquierda del bloque residual, la coordenada de la esquina superior izquierda del bloque de transformación es (Z, 0) para SVT-V o (0, Z) para SVT-H.

5 En un ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde w y h son el ancho y la altura del bloque residual, respectivamente, y $M1$ y $M2$ son números enteros predeterminados en el intervalo de 2 a 8. Se permiten más posiciones candidatas con un valor mayor de $M1$ o $M2$. En este ejemplo, tanto $M1$ como $M2$ se establecen como 8. Por lo tanto, el valor de P está entre 0 y 4. Las posiciones candidatas se ilustran en la figura 24.

10 En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde $Th1$ y $Th2$ son números enteros predefinidos que especifican un tamaño del escalón mínimo. $Th1$ y $Th2$ son números enteros no menores que 2. En este ejemplo, $Th1$ y $Th2$ se establecen como 4, y $M1$ y $M2$ se establecen como 8. En este ejemplo, los diferentes tamaños del bloque pueden tener un número diferente de posiciones candidatas. Por ejemplo, cuando $w=8$, están disponibles 2 posiciones candidatas para elegir (como se ilustra en la figura 24(A) y la figura 24(E)); cuando $w=16$, están disponibles 3 posiciones candidatas para elegir (como se ilustra en la figura 24(A), la figura 24(C) y la figura 24(E)); cuando $w>16$, hay 5 posiciones disponibles para elegir.

20 En otro ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde $M1$ y $M2$ se establecen como 4. En consecuencia, se permiten 3 posiciones candidatas.

En otro ejemplo, el CPSS se calcula como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde $M1$ y $M2$ se establecen como 2. En consecuencia, se permiten 2 posiciones candidatas.

25 En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde $T1$ y $T2$ se establecen como 2, $M1$ se establece como 8 si $w \geq h$ o se establece como 4 si $w < h$, y $M2$ se establece como 8 si $h \geq w$ o se establece como 4 si $h < w$. En este caso, el número de posiciones candidatas para SVT-H o SVT-V depende además de la relación de aspecto del bloque residual.

30 En otro ejemplo, el CPSS se calcula como $s = \max(w/M1, Th1)$ para SVT-V o $s = \max(h/M2, Th2)$ para SVT-H, donde los valores de $M1$, $M2$, $Th1$ y $Th2$ se derivan de una estructura de sintaxis de alto nivel en el flujo de bits (por ejemplo, el conjunto de parámetros de secuencia). $M1$ y $M2$ pueden compartir el mismo valor analizado de un elemento de sintaxis, y $Th1$ y $Th2$ pueden compartir el mismo valor analizado de otro elemento de sintaxis.

40 El índice de posición P puede binarizarse en uno o más contenedores usando código unario truncado. Por ejemplo, si el valor P está en el intervalo de 0 a 4, los valores P 0, 4, 2, 3 y 1 se binarizan como 0, 01, 001, 0001 y 0000, respectivamente; si el valor P está en el intervalo de 0 a 1, los valores P 0 y 1 se binarizan como 0 y 1, respectivamente.

Etapa 3. Analizar los coeficientes de transformación del bloque de transformación en función del tamaño del bloque de transformación.

45 Este es un proceso comúnmente usado en decodificación de vídeo, tal como el análisis de los coeficientes de transformación en HEVC o H.264/codificación de vídeo avanzada (AVC, por sus siglas en inglés). Los coeficientes de transformación pueden codificarse usando codificación de longitud de ejecución o codificarse de forma más sofisticada como un conjunto de grupos de coeficientes de transformación.

50 La Etapa 3 puede llevarse a cabo antes de la Etapa 2.

Etapa 4. Reconstruir el bloque residual R_0 en función de los coeficientes de transformación y la posición del bloque de transformación.

55 Se aplican cuantificación inversa y transformación inversa de tamaño $w_t \times h_t$ a los coeficientes de transformación para recuperar muestras residuales. El tamaño de las muestras residuales es $w_t \times h_t$, que es igual que el tamaño del bloque de transformación. La transformación inversa puede ser una transformación de coseno discreta inversa (DCT) o una transformación de seno discreta inversa (DST).

60 Las muestras residuales se asignan a una región correspondiente dentro del bloque residual R_0 , según la posición del bloque de transformación, y las muestras restantes dentro del bloque residual se establecen como cero. Por ejemplo, si se usa SVT-V y el número de posiciones candidatas es 5 y el índice de posición es 4, las muestras residuales reconstruidas se asignan a la región A en la figura 24(E) y la región de tamaño $(w/2) \times h$ a la izquierda de la región A tiene residual cero.

Después de realizar las Etapas 1 a 4, el bloque residual reconstruido se puede componer con un bloque de predicción para generar las muestras reconstruidas en una unidad de codificación. Posteriormente, se puede aplicar un proceso de filtrado a las muestras reconstruidas, tal como el filtro de desbloqueo y el procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC.

Implementación 2

En otra implementación, la información de posición de transformación se deriva de los coeficientes de transformación del bloque residual. Un bloque residual R_0 de tamaño $w \times h$ puede reconstruirse mediante las siguientes etapas.

Etapas 1. Determinar el tamaño del bloque de transformación del bloque residual R_0 .

Se puede aplicar el mismo proceso descrito en la Etapa 1 de la Implementación 1.

Etapas 2. Analizar los coeficientes de transformación del bloque de transformación en función del tamaño del bloque de transformación.

Se puede aplicar el mismo proceso descrito en la Etapa 3 de la Implementación 1.

Etapas 3. Derivar la posición del bloque de transformación en función de la suma de los coeficientes de transformación.

En un ejemplo, la suma de los coeficientes de transformación se calcula sumando todos los coeficientes de transformación del componente Y. En otro ejemplo, la suma de los coeficientes de transformación se calcula sumando todos los coeficientes de transformación de los componentes Y, U y V. En otro ejemplo, la suma de los coeficientes de transformación se calcula sumando los primeros N coeficientes de transformación del componente Y en orden de escaneo de los coeficientes (por ejemplo, usando un escaneo en zig-zag), donde N puede establecerse como 16 o 32.

El índice P de posición (que indica la posición del bloque de transformación, como se describe en la Implementación 1) se deriva de la suma de los coeficientes St de transformación y el número de posiciones candidatas N_t asociadas con el bloque de transformación, de acuerdo con la siguiente fórmula:

$$P = St \% N_t$$

donde % denota una operación módulo que encuentra el resto después de dividir St por N_t .

Más específicamente, en un ejemplo, los parámetros M_1 y M_2 se establecen como 4 y hay 3 posiciones candidatas para SVT-H o SVT-V (es decir, N_t para SVT-H o SVT-V es 3). La P se calcula como el resto de St dividido por 3. En otro ejemplo, los parámetros M_1 y M_2 se establecen como 2, y hay 2 posiciones candidatas para SVT-H o SVT-V (es decir, N_t para SVT-H o SVT-V es 2). La P se calcula como el resto de St dividido por 2.

En otro ejemplo, el índice de posición puede decidirse parcialmente mediante la suma de coeficientes de transformación. Los parámetros M_1 y M_2 se establecen como 8 y hay 5 posiciones candidatas para SVT-H o SVT-V. Si $St \% 3$ es igual a 0, P se determina como 0; si $St \% 3$ es igual a 1, P se determina como 4; de lo contrario, el valor de P está entre 1, 2 y 3 (pero aún no se ha decidido). Luego, se analiza un elemento de sintaxis que tiene tres valores posibles para determinar si P es igual a 1 o 2 o 3 en función del valor del elemento de sintaxis (por ejemplo, P se establece como 1, 2 o 3 si el valor del elemento de sintaxis es 0, 1 o 2).

Puede haber alguna verificación de condiciones para determinar si la información del índice de posición está oculta en la suma de coeficientes. Por ejemplo, cuando el número de coeficientes de transformación es menor que un umbral (por ejemplo, 4 o 6) o la distancia entre el primer y el último coeficientes es menor que un umbral (por ejemplo, 5 o 6), el índice de posición no se basa en la suma de coeficientes y se señala explícitamente, y el proceso de análisis en la Implementación 1 se aplica para determinar el índice de posición.

Etapas 4. Reconstruir el bloque residual R_0 en función de los coeficientes de transformación y la posición del bloque de transformación.

Se puede aplicar el mismo proceso descrito en la Etapa 4 de la Implementación 1.

Implementación 3

En otra implementación, el tipo de SVT se deriva en función de un primer grupo de coeficientes de transformación, y la información de posición se deriva de un segundo grupo de coeficientes de transformación del bloque residual.

5 Un bloque residual R_0 de tamaño $w \times h$ puede reconstruirse mediante las siguientes etapas.

Etapa 1. Determinar el tamaño del bloque de transformación del bloque residual R_0 , mediante la Etapa 1.1 y la Etapa 1.2.

10 Etapa 1.1. Determinar el uso de SVT. Esta es igual que la Etapa 1.1 en la Realización 1.

Etapa 1.2. Analizar un primer grupo de coeficientes de transformación y determinar el tipo de SVT en función de la suma del primer grupo de coeficientes de transformación.

15 El primer grupo de coeficientes de transformación puede estar en una región de tamaño $w_{t1} \times h_{t1}$ en la parte superior izquierda del bloque de transformación (correspondiente a los coeficientes de transformación de frecuencia más baja), como se ilustra mediante la región 1 sólida en la figura 25. En un ejemplo, w_{t1} y h_{t1} se establecen ambos como 4. En otro ejemplo, $w_{t1} = \max(w/8, 4)$ y $h_{t1} = \max(h/8, 4)$.

20 Si la suma de los coeficientes del primer grupo es un número par, se usa SVT-V (o SVT-H); de lo contrario, se usa SVT-H (o SVT-V).

Etapa 2. Analizar los coeficientes de transformación restantes en el bloque de transformación en función del tamaño del bloque de transformación.

25

El orden de escaneo de los coeficientes se deriva en función del ancho y la altura del bloque de transformación. Cuando se analizan los coeficientes de transformación, los coeficientes en las posiciones cubiertas por el primer grupo de coeficientes de transformación se omiten y se asignan para ser el primer grupo de coeficientes de transformación.

30

Etapa 3. Derivar la posición del bloque de transformación en función de la suma de un segundo grupo de los coeficientes de transformación.

35 Se usa un segundo grupo de coeficientes de transformación para calcular una suma. En función de la suma, se deriva el índice de posición, donde se puede usar el método descrito en la Implementación 2. El segundo grupo de coeficientes de transformación puede estar en una región de tamaño $w_{t2} \times h_{t2}$ en la parte superior izquierda del bloque de transformación separada del primer grupo de coeficientes de transformación, como se ilustra en la región 2 punteada en la figura 25. En un ejemplo, $w_{t2} = w_t$ y $h_{t2} = h_t$, es decir, todos los coeficientes excepto los coeficientes del primer grupo pertenecen al segundo grupo. En otro ejemplo, $w_{t2} = \max(w_t/2, w_{t1})$ y $h_{t2} = \max(h_t/2, h_{t1})$.

40

Etapa 4. Reconstruir el bloque residual R_0 en función de los coeficientes de transformación y la posición del bloque de transformación. Se puede usar el procesamiento de la Etapa 4 de la Implementación 1.

45 Implementación 4

En otra implementación, el número de posiciones candidatas para un bloque residual que usa SVT se deriva en función de la distribución del coeficiente de transformación.

50 Un bloque residual R_0 de tamaño $w \times h$ puede reconstruirse mediante las siguientes etapas.

Etapa 1. Determinar el tamaño del bloque de transformación del bloque residual R_0 .

55 Se puede aplicar el mismo proceso descrito en la Etapa 1 de la Implementación 1.

Etapa 2. Analizar los coeficientes de transformación del bloque de transformación en función del tamaño del bloque de transformación.

60 Se puede aplicar el mismo proceso descrito en la Etapa 3 de la Implementación 1.

Etapa 3. Derivar la posición del bloque de transformación en función de la distribución de los coeficientes de transformación.

65 En un ejemplo, el número de coeficientes de transformación distintos de cero N_{nz} se usa para decidir el número de posiciones candidatas. Si N_{nz} es menor que un umbral (por ejemplo, 4 o 5 u 8), el tamaño del escalón de la posición candidata se establece como $w/4$ para SVT-V y $h/4$ para SVT-H, y por lo tanto el número de

posiciones candidatas es 3; de lo contrario, el tamaño del escalón de la posición candidata se establece como $w/8$ para SVT-V y $h/8$ para SVT-H, por lo que el número de posiciones candidatas es 5.

5 En otro ejemplo, si los últimos coeficientes de transformación distintos de cero están dentro del área superior izquierda de 4×4 en el bloque de transformación, el tamaño del escalón de la posición candidata se establece como $w/4$ para SVT-V y $h/4$ para SVT-H, y por lo tanto el número de posiciones candidatas es 3; de lo contrario, el tamaño del escalón de la posición candidata se establece como $w/8$ para SVT-V y $h/8$ para SVT-H, y por lo tanto, el número de posiciones candidatas es 5.

10 En función del número de posiciones candidatas, el índice de posición se decodifica con un código unario truncado, siendo el valor más grande posible el número de posiciones candidatas menos uno.

Etapa 4. Reconstruir el bloque residual R_0 en función de los coeficientes de transformación y la posición del bloque de transformación. Se puede usar el procesamiento de la Etapa 4 de la Implementación 1.

15 La presente descripción proporciona los siguientes beneficios y ventajas con respecto a las soluciones conocidas.

20 Las posiciones candidatas para los bloques SVT se deciden por un tamaño del escalón relacionado con el ancho y la altura del bloque residual y se limitan a no más de 5, lo que reduce la información del bloque de transformación en comparación con [1] y logra un mejor equilibrio entre la sobrecarga de señalización y la eficacia de codificación residual.

25 La información de SVT está codificada de forma diferente a [1], es decir, está organizada como una información de uso de SVT, una información de tipo de SVT y una información de posición de SVT.

La información de posición de SVT puede derivarse en función de la suma de una pluralidad de coeficientes de transformación del bloque residual.

30 La información de tipo de SVT puede derivarse en función de la suma de una pluralidad de coeficientes de transformación del bloque residual.

35 Un experto en la técnica puede comprender claramente que, a efectos de una descripción breve y conveniente, la división de los módulos funcionales anteriores se toma como ejemplo ilustrativo. En una aplicación real, las funciones anteriores se pueden asignar a diferentes módulos funcionales e implementarse según se requiera, es decir, una estructura interna de un aparato se divide en diferentes módulos funcionales para implementar todas o algunas de las funciones descritas anteriormente. Para un proceso de funcionamiento detallado del sistema, el aparato y la unidad anteriores, consúltese un proceso correspondiente en las realizaciones del método anteriores, y los detalles no se describen de nuevo en la presente memoria.

40 En las varias realizaciones proporcionadas en la presente solicitud, se debe entender que el sistema, aparato y procedimiento descritos se pueden implementar de otras maneras. Por ejemplo, la realización de aparato descrita es simplemente ejemplar. Por ejemplo, la división en módulos o unidades es simplemente una división en funciones lógicas y puede ser otra división en una implementación real. Por ejemplo, una pluralidad de unidades o componentes pueden combinarse o integrarse en otro sistema, o algunas características pueden ignorarse o no realizarse. Además, los acoplamientos mutuos o acoplamientos directos o conexiones de comunicación que se muestran o se analizan, se pueden implementar usando otras interfaces. Los acoplamientos o conexiones de comunicación indirectos entre los aparatos o unidades se pueden implementar de forma electrónica, mecánica u otras.

50 Las unidades descritas como partes separadas pueden estar o no físicamente separadas, y las partes mostradas como unidades pueden ser o no unidades físicas, pueden estar ubicadas en una posición, o pueden estar distribuidas en una pluralidad de unidades de red. Algunas o todas las unidades se pueden seleccionar según las necesidades reales para lograr los objetivos de las soluciones de las realizaciones.

55 Además, las unidades funcionales en las realizaciones de la presente invención se pueden integrar en una unidad de procesamiento, o cada una de las unidades puede existir sola físicamente, o dos o más unidades se integran en una unidad. La unidad integrada se puede implementar en una forma de *hardware*, o se puede implementar en una forma de unidad funcional de *software*.

60 Cuando la unidad integrada se implementa en la forma de unidad funcional de *software* y se comercializa o se usa como un producto independiente, la unidad integrada se puede almacenar en un medio de almacenamiento legible por ordenador. en función de tal entendimiento, las soluciones técnicas de la presente invención esencialmente, o la parte que contribuye a la técnica anterior, o todas o alguna de las soluciones técnicas, se pueden implementar en la forma de un producto de *software*. El producto de *software* se almacena en un medio de almacenamiento e incluye varias instrucciones para dar instrucciones a un dispositivo informático (que

65

puede ser un ordenador personal, un servidor o un dispositivo de red) o un procesador (procesador) para realizar todas o alguna de las etapas de los métodos descritos en las realizaciones de la presente invención. El medio de almacenamiento anterior incluye: cualquier medio que pueda almacenar código de programa, tal como una unidad flash USB, un disco duro extraíble, una memoria de solo lectura (ROM, memoria de solo lectura), una memoria de acceso aleatorio (RAM, memoria de acceso aleatorio), un disco magnético o un disco óptico.

Las descripciones anteriores son solamente formas de implementación específicas de la presente invención, pero no pretenden limitar el alcance de protección de la presente invención. Cualquier variación o reemplazo fácilmente resuelto por un experto en la técnica dentro del alcance técnico de las reivindicaciones caerá dentro del alcance de protección de la presente invención. Por lo tanto, el alcance de protección de la presente invención estará sujeto al alcance de protección de las reivindicaciones.

REIVINDICACIONES

1. Un método de codificación de vídeo, que comprende:

5 obtener una unidad de codificación, CU;

determinar si se emplea SVT para la CU, en donde cuando se aplica SVT a la CU, la CU tiene solo una unidad de transformación residual, TU, y un tamaño de la TU residual es menor que un tamaño de la CU;

10 cuando se determina emplear SVT para la CU, determinar un tipo de SVT de la CU que especifica un modo de partición de TU de la CU, en donde se infiere que el resto de las TU de la CU no tienen residual;

dividir la CU en al menos dos TU en función del modo de partición de TU;

15 determinar una posición de SVT indica una posición de una TU residual entre las al menos dos TU;

determinar un tipo de transformación del bloque de transformación residual basándose en el tipo de SVT y la posición de SVT;

20 realizar un procesamiento de transformación en la TU residual para obtener coeficientes de transformación de la CU;

25 generar un flujo de bits incluye los coeficientes de transformación, un elemento de sintaxis especifica si se emplea SVT para la CU, un indicador especifica el tipo de SVT y un elemento de sintaxis especifica la posición de SVT; y

transmitir el flujo de bits a un decodificador de vídeo, en donde un modo de transformación de destino de la TU residual se selecciona de un conjunto que comprende los siguientes modos:

30 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU superior es menor que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8, en donde el tamaño de la TU superior se indica mediante $W \times (H/4)$ y el tamaño de la TU inferior se indica mediante $W \times (3H/4)$;

35 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU inferior es menor que un tamaño de una TU superior, y la TU residual es la TU inferior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU inferior se indica mediante $W \times (H/4)$ y el tamaño de la TU superior se indica mediante $W \times (3H/4)$;

40 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, y la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal es DCT-8 y una transformación vertical es DST-7, en donde el tamaño de la TU izquierda se indica mediante $(W/4) \times H$ y el tamaño de la TU derecha se indica mediante $(3W/4) \times H$;

45 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU derecha se indica mediante $(W/4) \times H$ y el tamaño de la TU izquierda se indica mediante $(3W/4) \times H$;

50 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8;

55 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU inferior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7;

60 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la posición de la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7; y

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7.

5

2. Un codificador de vídeo, que comprende:

una unidad configurada para obtener una unidad de codificación, CU;

10 una unidad configurada para determinar si se emplea SVT para la CU, en donde cuando se aplica SVT a la CU, la CU tiene solo una unidad de transformación residual, TU, y un tamaño de la TU residual es menor que un tamaño de la CU;

15 una unidad configurada para determinar un tipo de SVT de la CU que especifica un modo de partición de TU de la CU cuando se determina emplear SVT para la CU, en donde se infiere que el resto de las TU de la CU no tienen residual;

una unidad configurada para dividir la CU en al menos dos TU basándose en el modo de partición de TU;

20 una unidad configurada para determinar una posición de SVT indica una posición de una TU residual entre las al menos dos TU;

una unidad configurada para determinar un tipo de transformación del bloque de transformación residual basándose en el tipo de SVT y la posición de SVT;

25

una unidad configurada para realizar un procesamiento de transformación en la TU residual para obtener coeficientes de transformación de la CU;

30 una unidad configurada para generar un flujo de bits incluye los coeficientes de transformación, un elemento de sintaxis especifica si se emplea SVT para la CU, un indicador especifica el tipo de SVT y un elemento de sintaxis especifica la posición de SVT; y

una unidad configurada para transmitir el flujo de bits a un decodificador de vídeo; en donde un modo de transformación de destino de la TU residual se selecciona de un conjunto que comprende los siguientes modos:

35

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU superior es menor que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8, en donde el tamaño de la TU superior se indica mediante $W \times (H/4)$ y el tamaño de la TU inferior se indica mediante $W \times (3H/4)$;

40

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU inferior es menor que un tamaño de una TU superior, y la TU residual es la TU inferior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU inferior se indica mediante $W \times (H/4)$ y el tamaño de la TU superior se indica mediante $W \times (3H/4)$;

45

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, y la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal es DCT-8 y una transformación vertical es DST-7, en donde el tamaño de la TU izquierda se indica mediante $(W/4) \times H$ y el tamaño de la TU derecha se indica mediante $(3W/4) \times H$;

50

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU derecha se indica mediante $(W/4) \times H$ y el tamaño de la TU izquierda se indica mediante $(3W/4) \times H$;

55

60 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8;

60

65 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU inferior, el tipo

65

de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7;

5 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la posición de la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7; y

10 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7.

3. Un aparato, que comprende:

15 uno o más procesadores; y un medio de almacenamiento legible por ordenador no transitorio acoplado a los procesadores y que almacena la programación para su ejecución por los procesadores, en donde la programación, cuando la ejecutan los procesadores, configura el aparato para llevar a cabo el método según la reivindicación 1.

4. Un flujo de bits, que comprende:

20 coeficientes de transformación obtenidos realizando un procesamiento de transformación en una unidad de transformación residual, TU, en donde la TU residual es la única TU residual de una unidad de codificación, CU, en donde se emplea SVT para la CU;

25 un elemento de sintaxis especifica si se debe emplear SVT para la CU;

un indicador especifica un tipo de SVT de la CU que especifica un modo de partición de TU de la CU; y

30 un elemento de sintaxis especifica una posición de SVT que indica una posición de la TU residual entre al menos dos TU obtenidas al dividir la CU basándose en el modo de partición de TU, en donde se infiere que el resto de las TU de la CU no tienen residual;

35 en donde el tipo de SVT y la posición de SVT se usan para determinar un tipo de transformación de la TU residual;

en donde un modo de transformación de destino de la TU residual se selecciona de un conjunto que comprende los siguientes modos:

40 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU superior es menor que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8, en donde el tamaño de la TU superior se indica mediante $W \times (H/4)$ y el tamaño de la TU inferior se indica mediante $W \times (3H/4)$;

45 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, un tamaño de una TU inferior es menor que un tamaño de una TU superior, y la TU residual es la TU inferior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU inferior se indica mediante $W \times (H/4)$ y el tamaño de la TU superior se indica mediante $W \times (3H/4)$;

50 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es menor que un tamaño de una TU derecha, y la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal es DCT-8 y una transformación vertical es DST-7, en donde el tamaño de la TU izquierda se indica mediante $(W/4) \times H$ y el tamaño de la TU derecha se indica mediante $(3W/4) \times H$;

55 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU derecha es menor que un tamaño de una TU izquierda, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7, en donde el tamaño de la TU derecha se indica mediante $(W/4) \times H$ y el tamaño de la TU izquierda se indica mediante $(3W/4) \times H$;

60 en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU superior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DCT-8;

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU superior es igual que un tamaño de una TU inferior, y la TU residual es la TU inferior, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7;

5

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la posición de la TU residual es la TU izquierda, el tipo de transformación es que una transformación horizontal de la TU residual es DCT-8 y una transformación vertical de la TU residual es DST-7; y

10

en donde, cuando el modo de partición de TU especifica que la CU debe dividir la CU en 2 TU, en donde un tamaño de una TU izquierda es igual que un tamaño de una TU derecha, y la TU residual es la TU derecha, el tipo de transformación es que una transformación horizontal es DST-7 y una transformación vertical es DST-7.

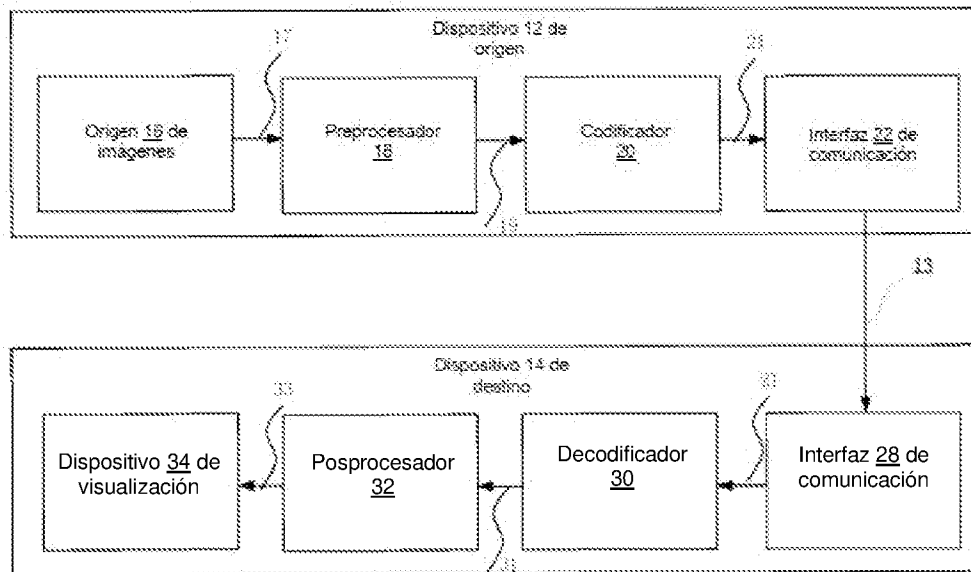


Fig. 1A

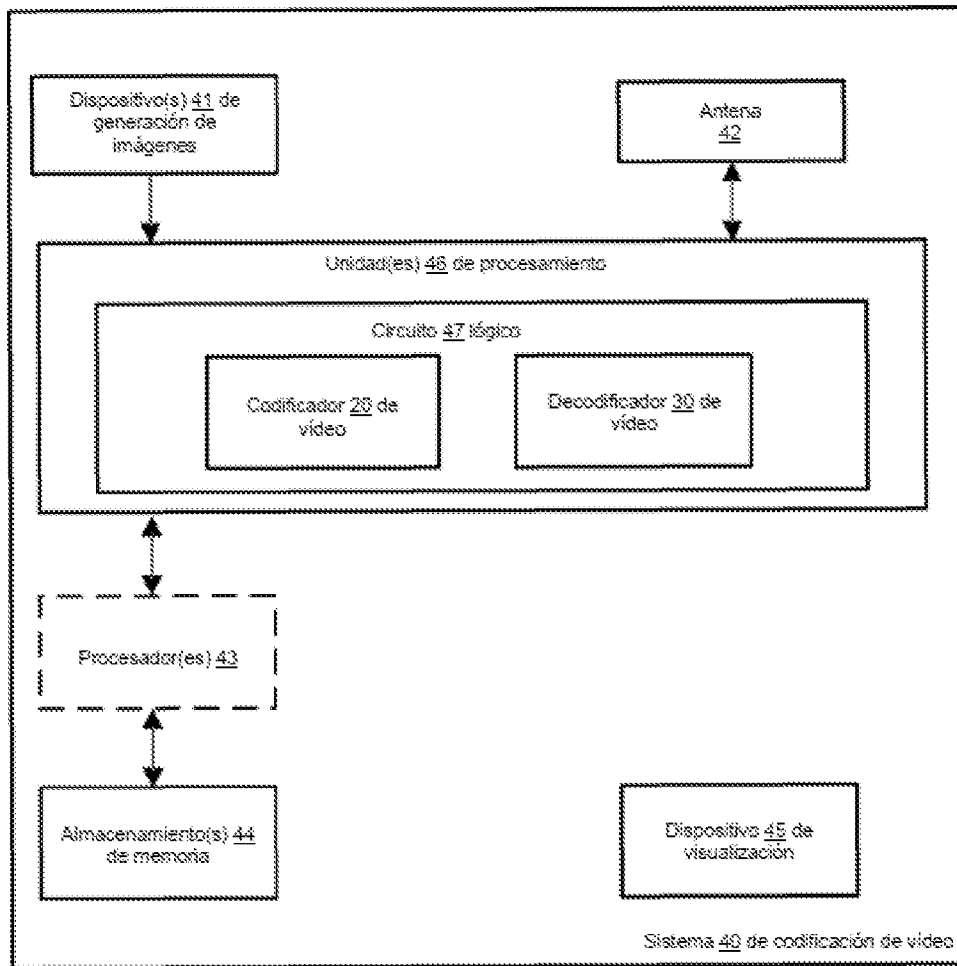


Fig. 1B

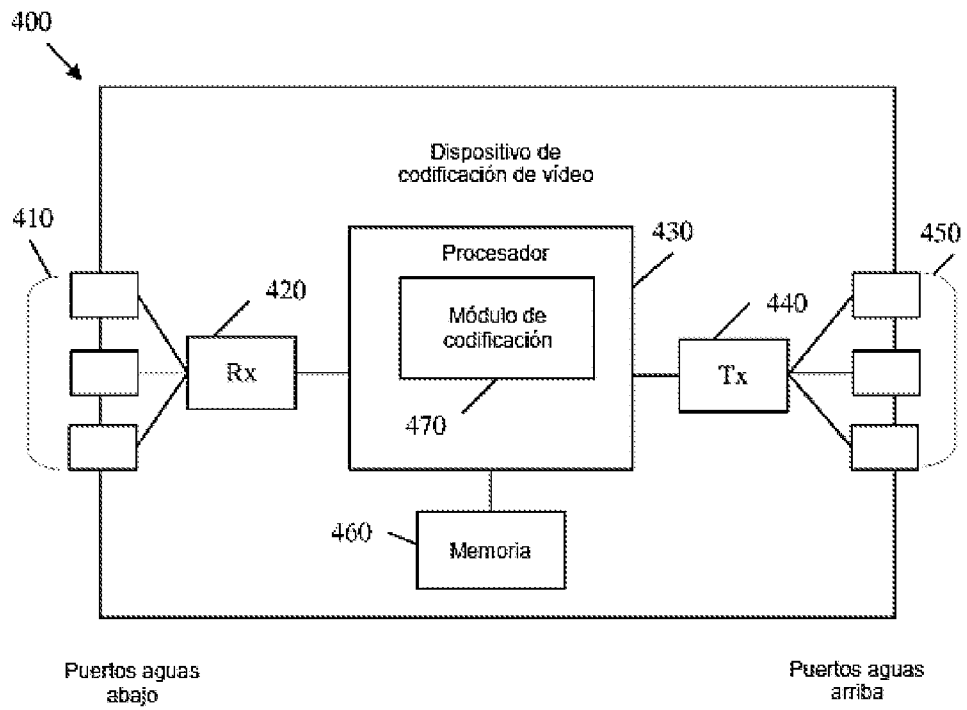


Fig. 4

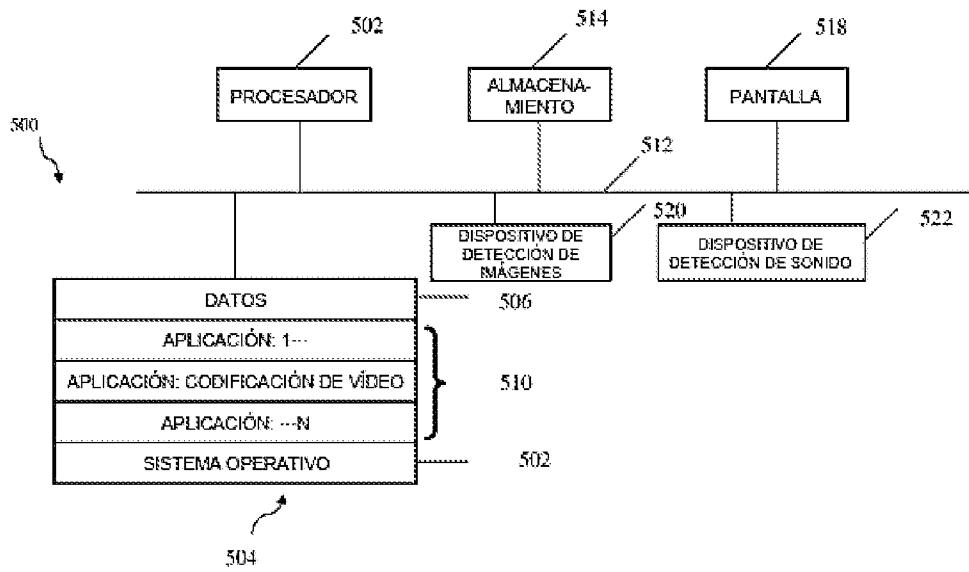


Fig. 5

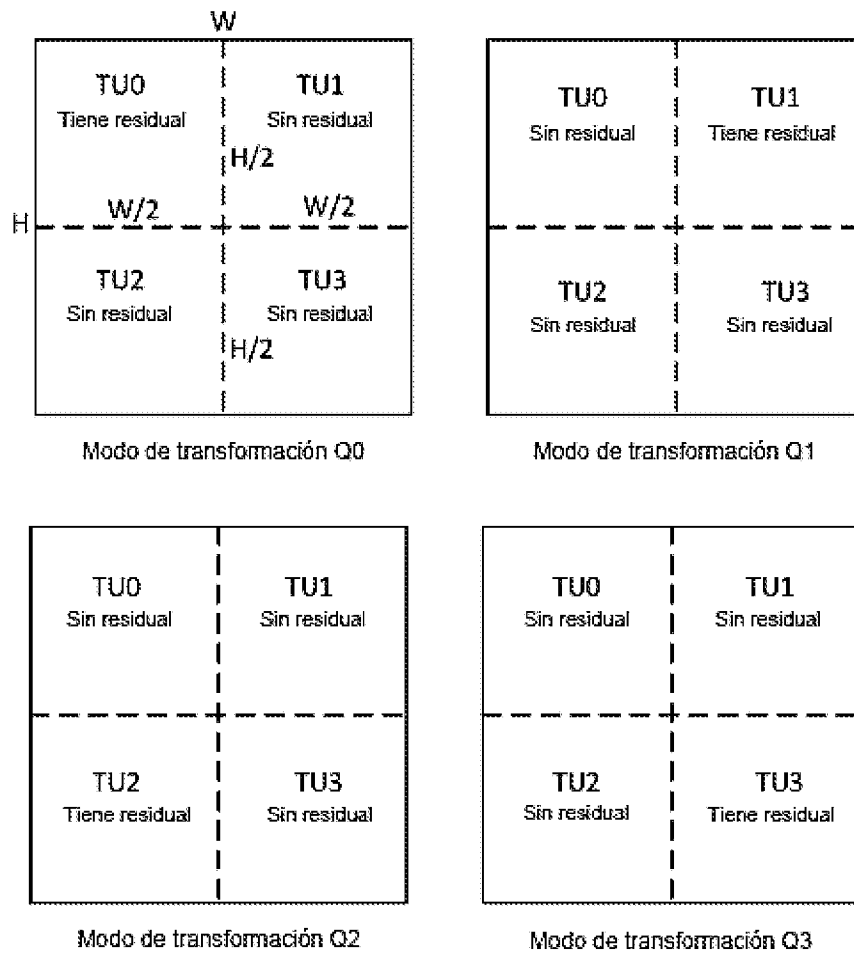


Fig. 6

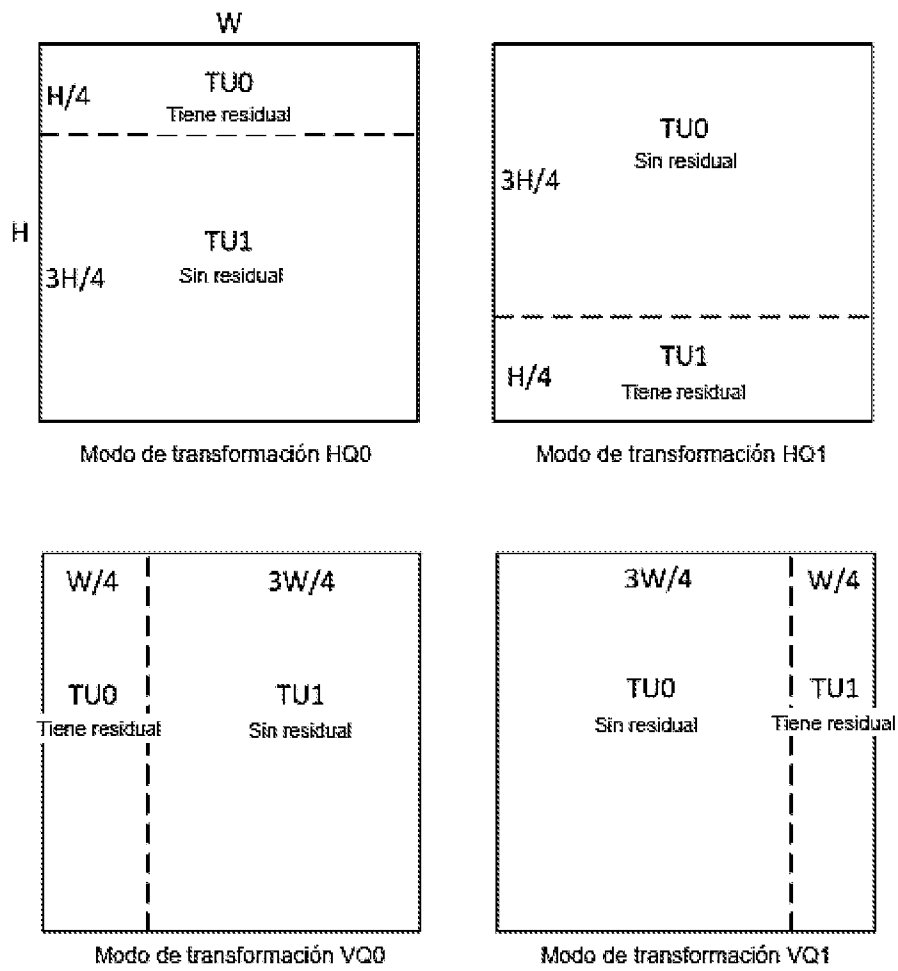


Fig. 7

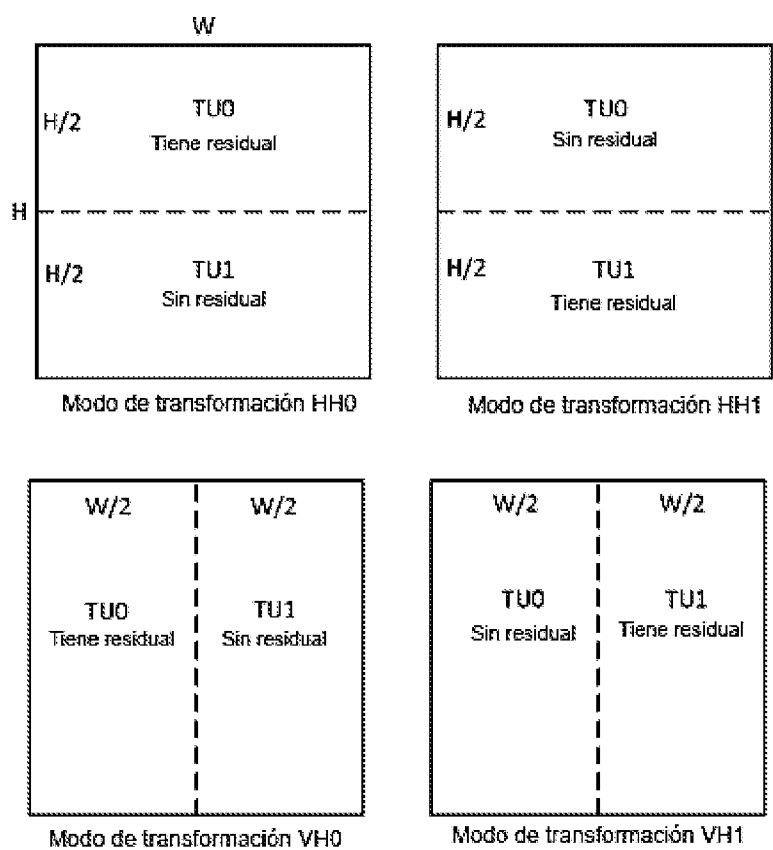


Fig. 8

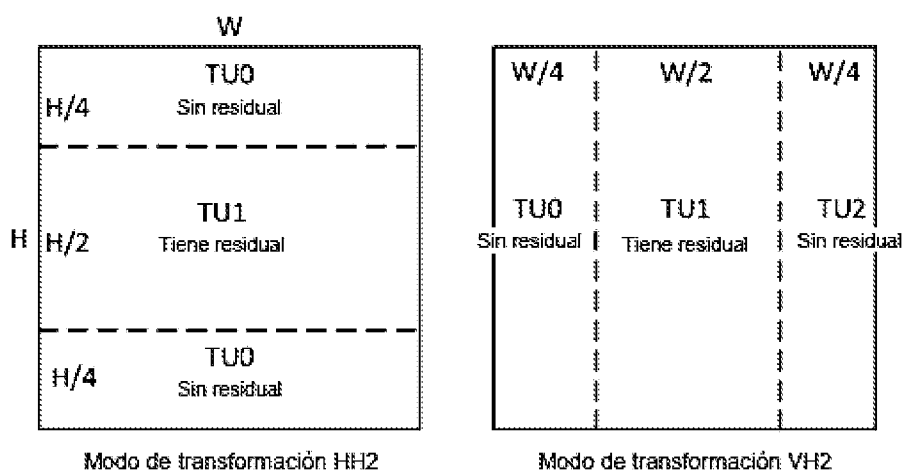


Fig. 9

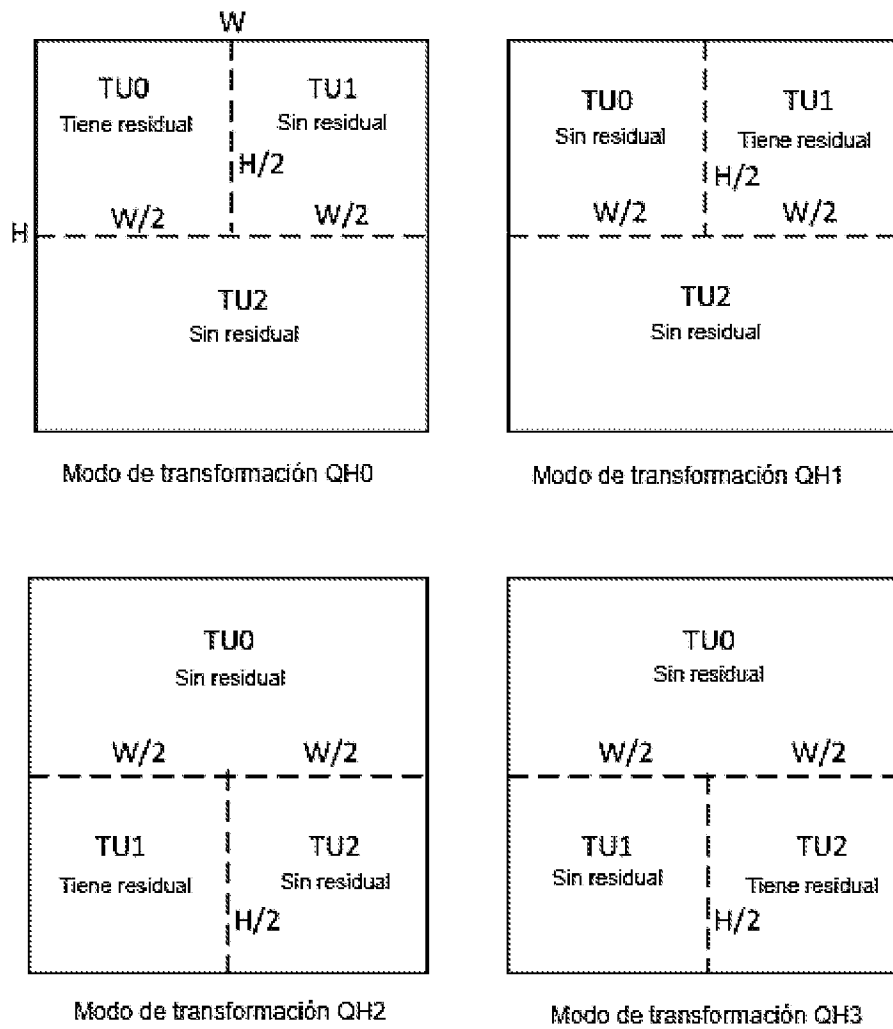


Fig. 10

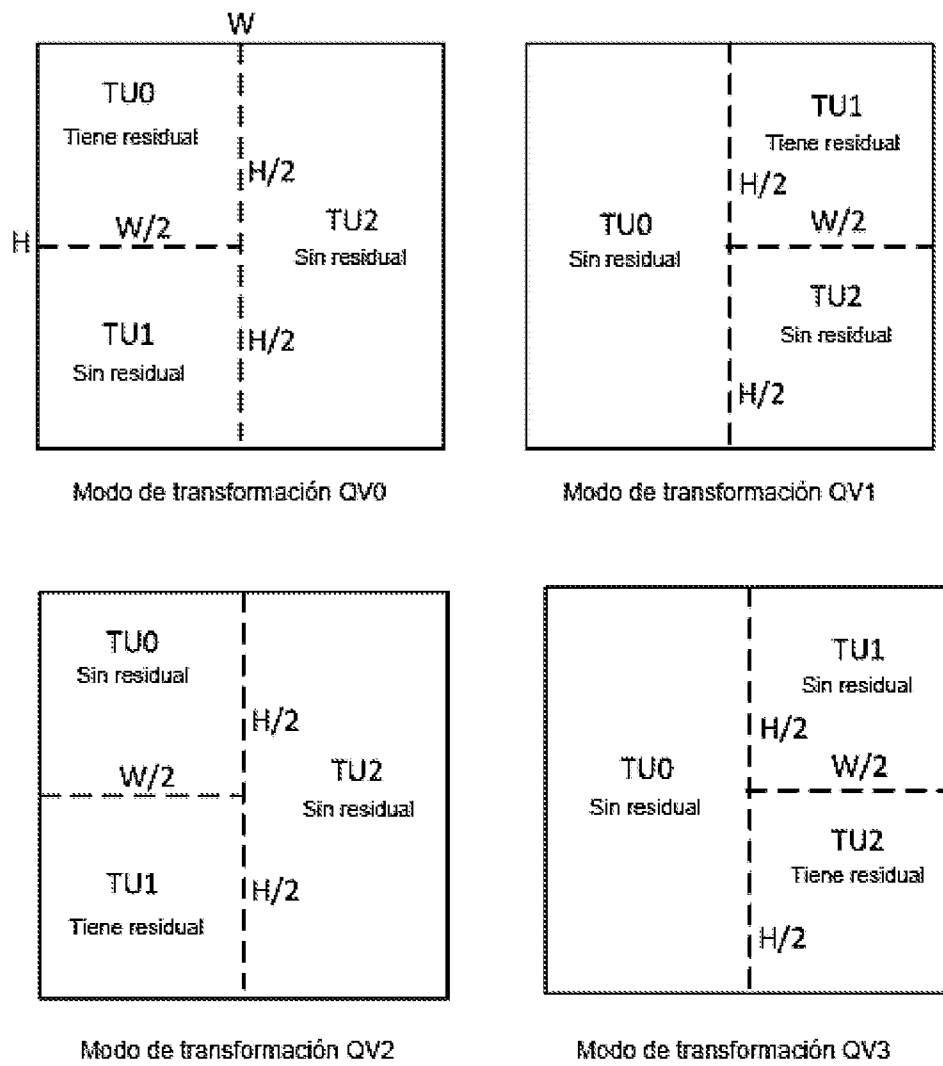


Fig. 11

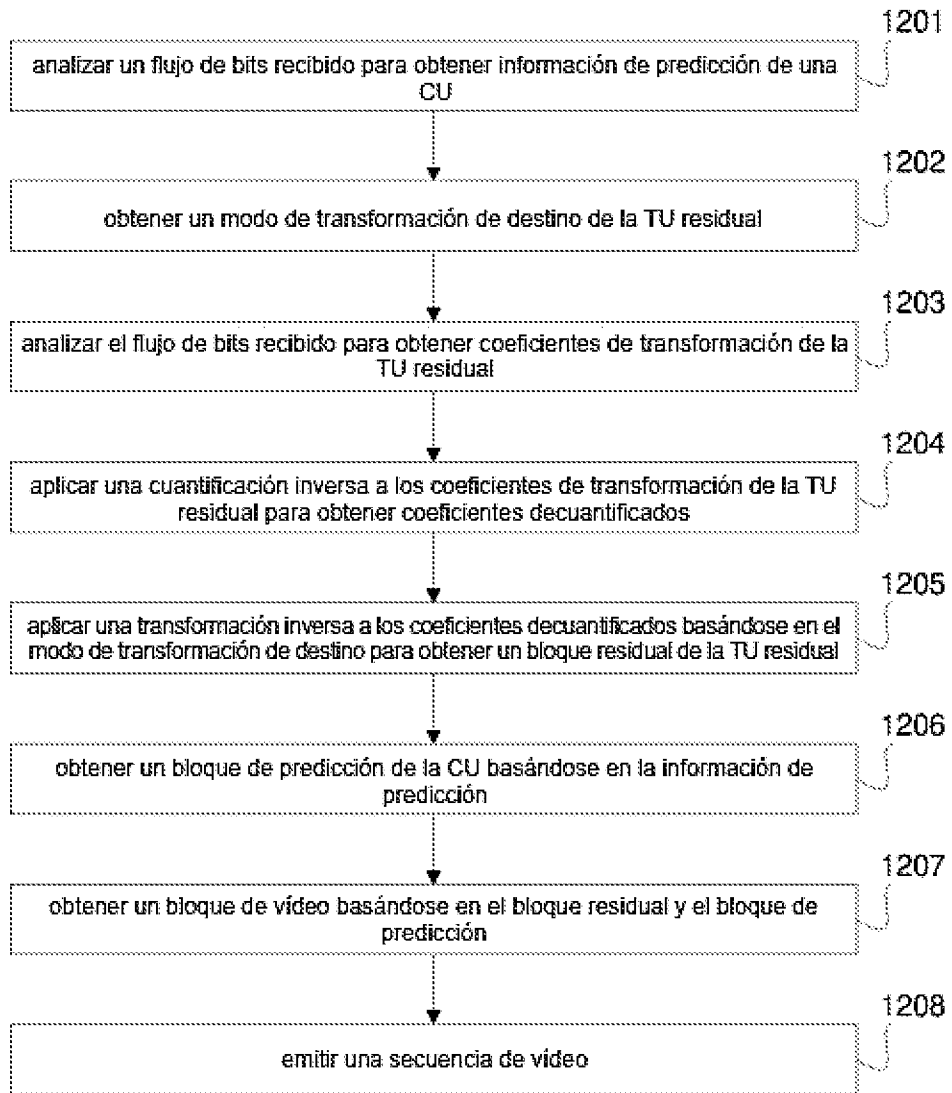


Fig. 12

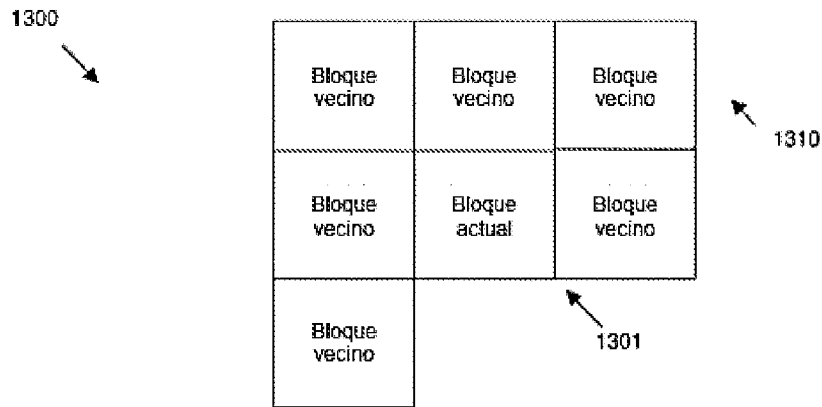


Fig. 13

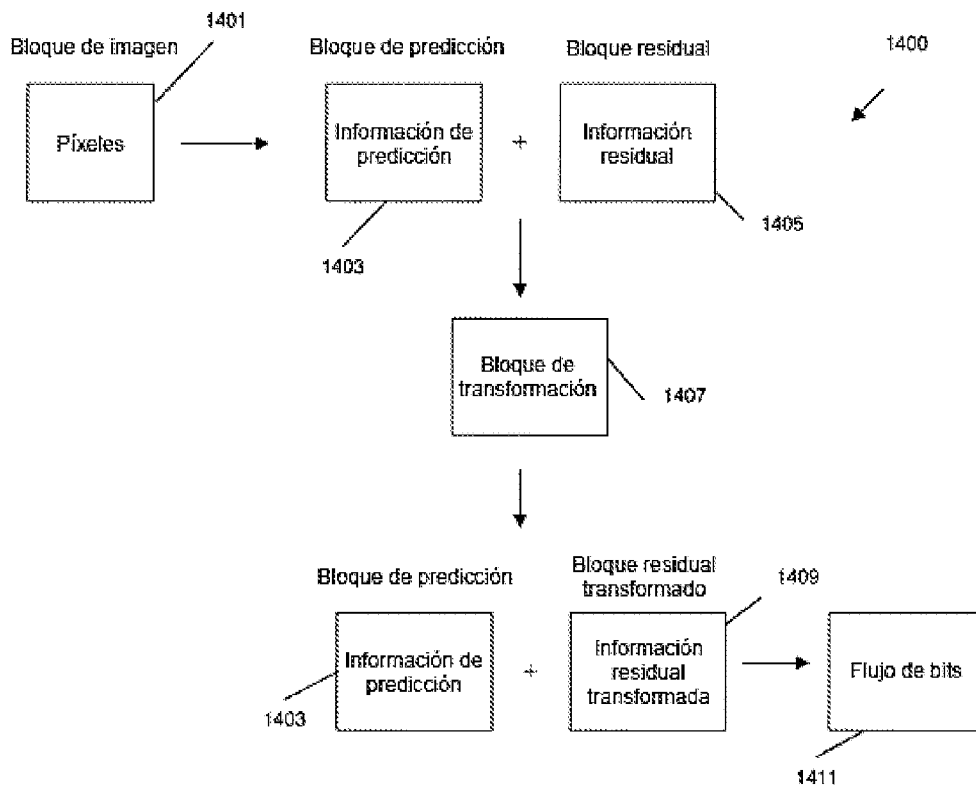


Fig. 14

1500

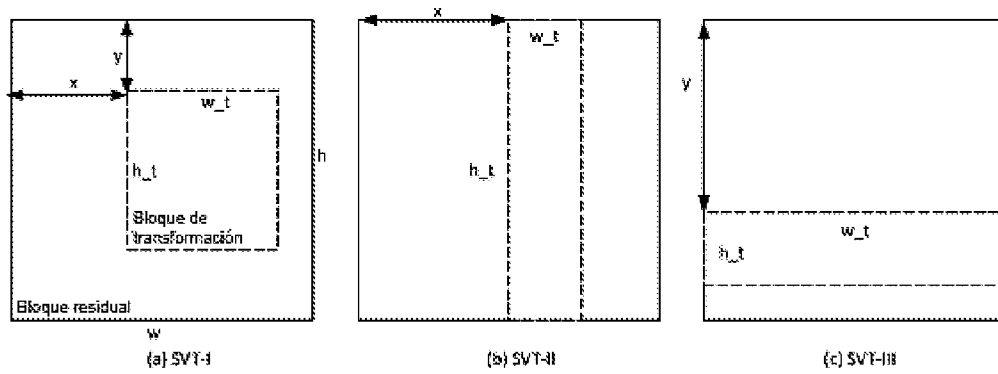


Fig. 15

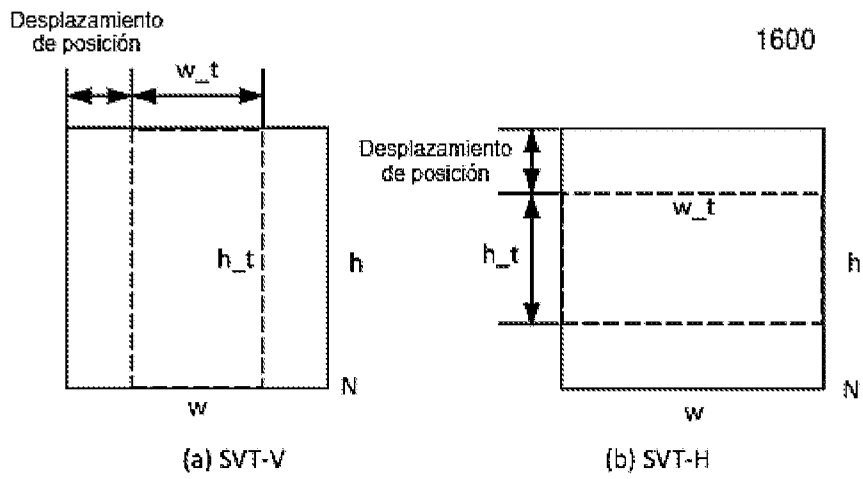


Fig. 16

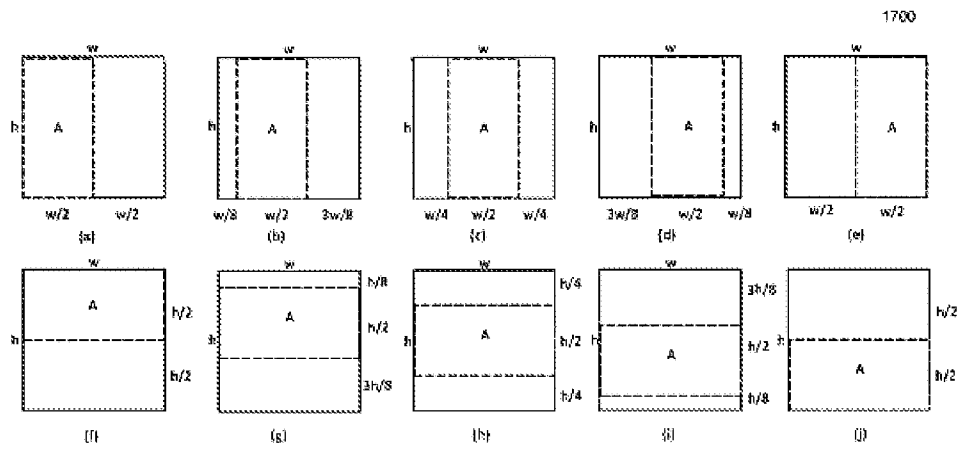


Fig. 17

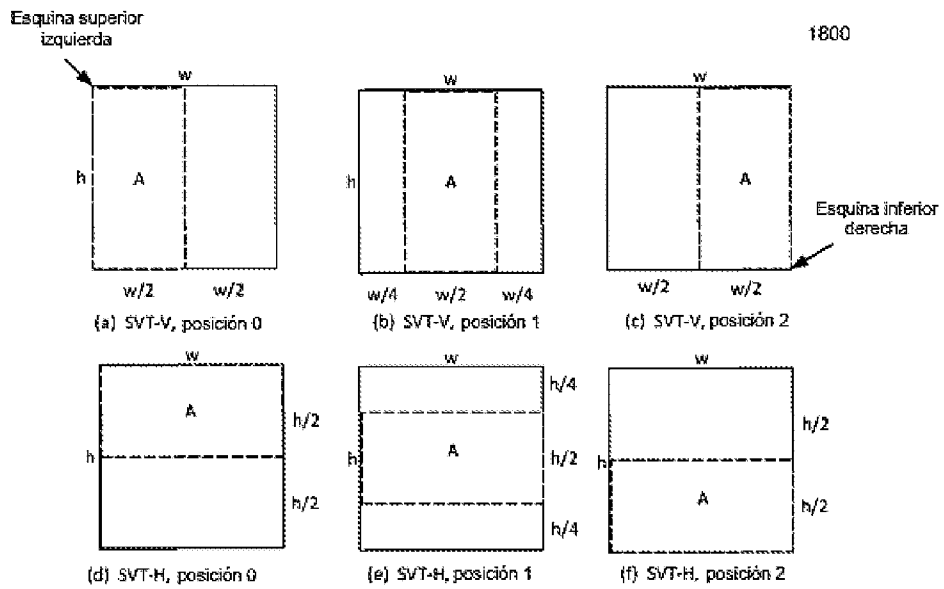


Fig. 18

1900

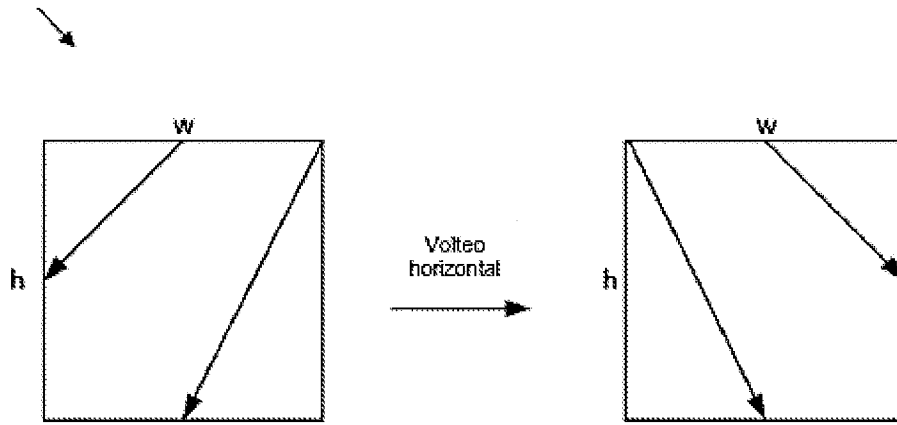


Fig. 19

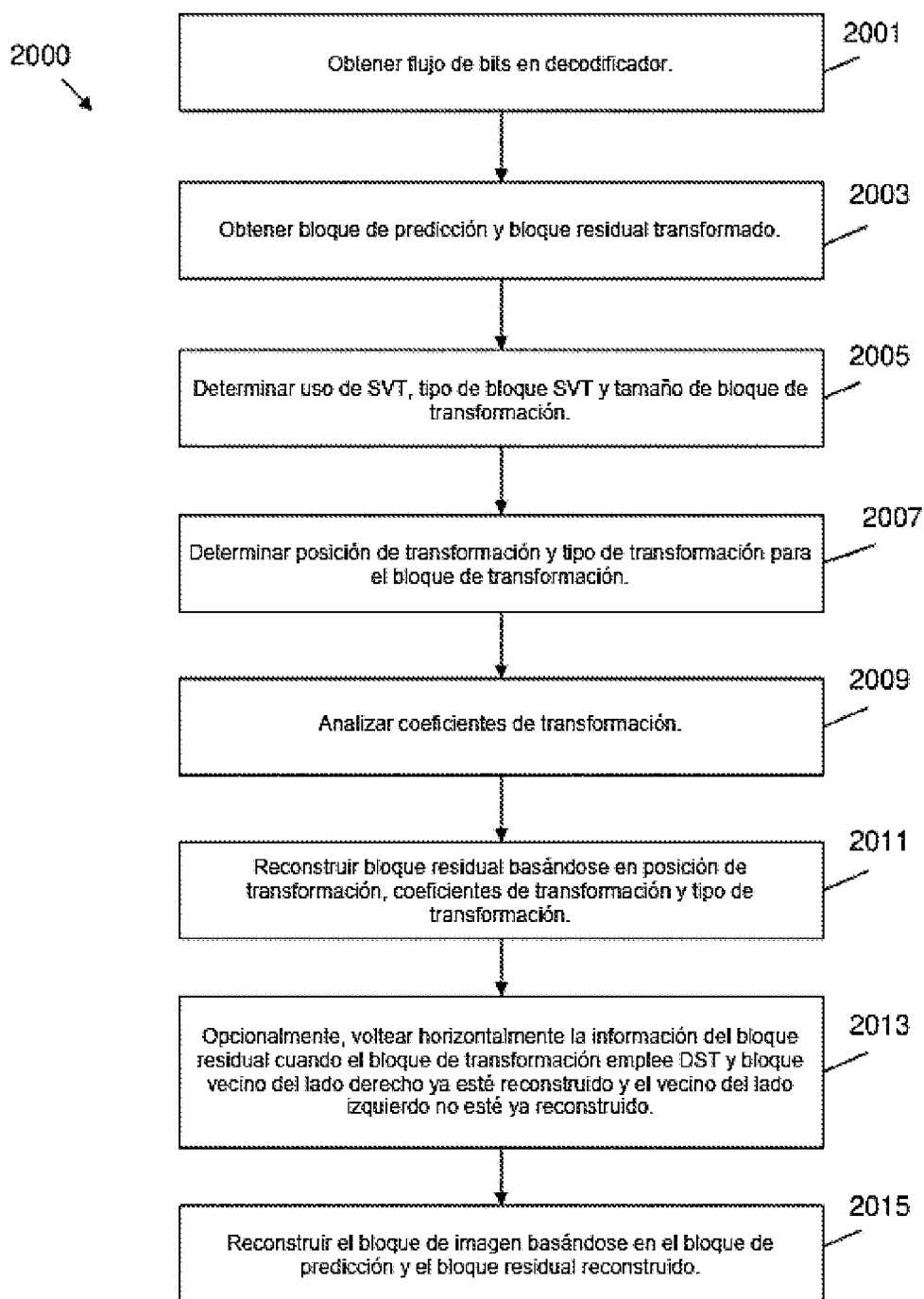


Fig. 20

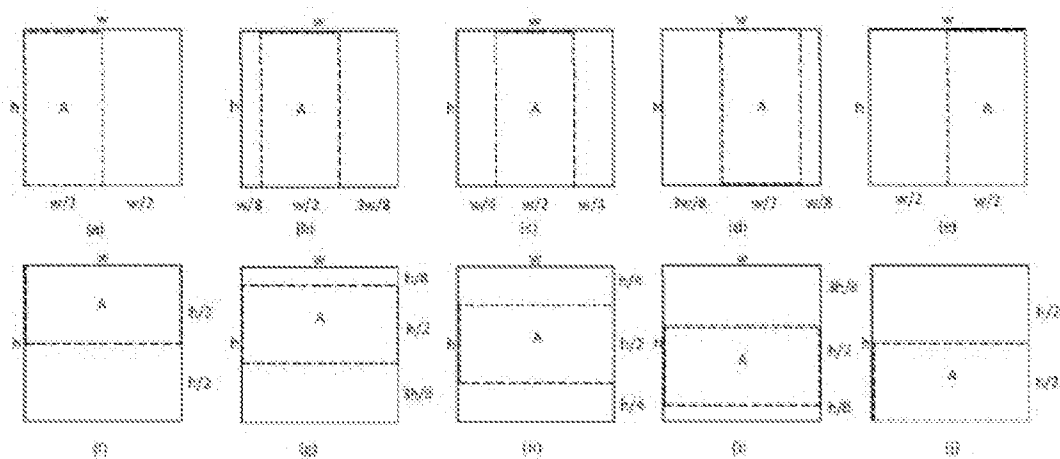


Fig. 21

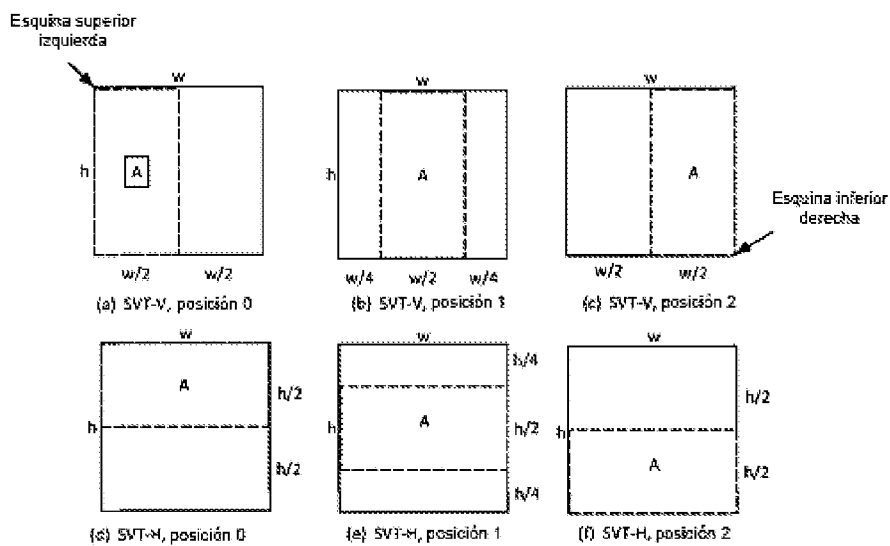


Fig. 22

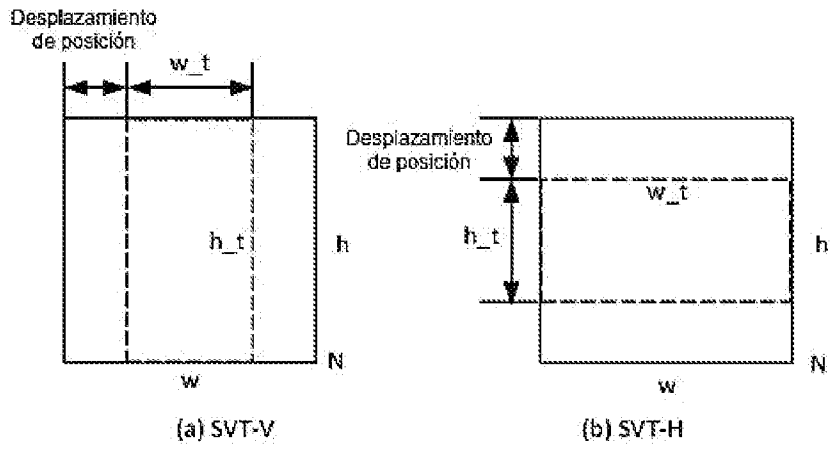


Fig. 23

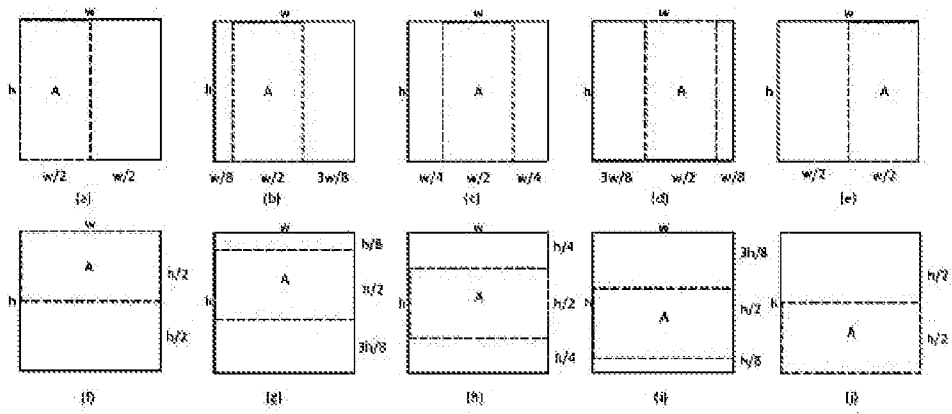


Fig. 24

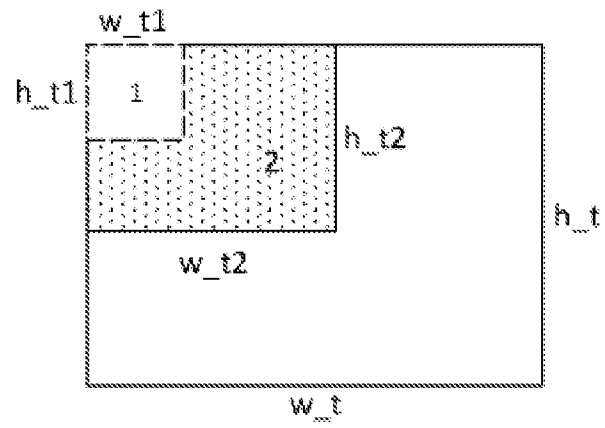


Fig. 25