

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 977 659**

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

86 Fecha de presentación y número de la solicitud internacional: **08.10.2018** **PCT/CN2018/109361**

87 Fecha y número de publicación internacional: **25.04.2019** **WO19076206**

96 Fecha de presentación y número de la solicitud europea: **08.10.2018** **E 18868850 (1)**

97 Fecha y número de publicación de la concesión europea: **07.02.2024** **EP 3677032**

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
06.10.2018 WO PCT/CN2018/109254

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
28.08.2024

73 Titular/es:

HUAWEI TECHNOLOGIES CO., LTD. (100.0%)
Huawei Administration Building, Bantian,
Longgang District
Shenzhen, Guangdong 518129, CN

72 Inventor/es:

ZHAO, YIN;
YANG, HAITAO y
CHEN, JIANLE

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 977 659 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 CAMPO TÉCNICO

La presente invención está relacionada con el campo de comunicaciones, y en particular, con un método y un aparato de codificación.

10 ANTECEDENTES

La cantidad de datos de vídeo necesarios para representar incluso un vídeo relativamente corto pueden ser sustanciales, que puede tener como resultado dificultades cuando los datos se van a transmitir o comunicar de otro modo por una red de comunicaciones con capacidad limitada de ancho de banda. Así, los datos de vídeo generalmente se comprimen antes de ser comunicados por redes de telecomunicaciones diurnas por módem. El tamaño de un vídeo también podría ser una cuestión cuando el vídeo se almacena en un dispositivo de almacenamiento porque 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 almacenamiento, disminuyendo de ese modo la cantidad de datos necesarios para representar imágenes de vídeo digital. Los datos comprimidos se reciben entonces en el destino por un dispositivo de descompresión de vídeo que decodifica los datos de vídeo.

Con recursos de red limitados y demandas siempre crecientes de calidad de vídeo más alta, son deseables técnicas de compresión y descompresión mejoradas que mejoren la relación de compresión con poco o ningún sacrificio en la calidad de imagen.

El documento "Video coding using Variable Block-Size Spatially Varying Transforms" (de Cixun Zhang et. al., abril 2009) divulga un método de Transformada Espacialmente Variable de Tamaño de Bloque Variable (VBSVT), que selecciona y codifica un bloque de 16x4/4x16 dentro de un macrobloque 16x16 con una transformada correspondiente 16x4/4x16.

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 transformada dependiente de frontera para interpredicción, en la que se seleccionan diferentes transformadas (DST-VII/DCT-IV o DCT-II) dependiendo de si la frontera de la Unidad de Transformada (TU) es una frontera de una Unidad de Predicción (PU).

35 COMPENDIO

Realizaciones de la presente invención proporcionan un método y un aparato de codificación, que puede realizar una apropiada adjudicación de bits por cuantización para coeficientes espectrales de una señal de audio, mejorando de ese modo la calidad de una señal obtenida por un decodificador por medio de decodificación.

La invención se define en las reivindicaciones independientes. Rasgos adicionales de la invención se proporcionan en las reivindicaciones dependientes. A continuación, partes de la descripción y los dibujos que hacen referencia a realizaciones, que no están cubiertas por las reivindicaciones no se presentan como realizaciones de la invención, sino como ejemplos útiles para entender la invención.

Para lograr el objetivo anterior, se usan las siguientes soluciones técnicas:

Según un primer aspecto, se proporciona un método de decodificación de vídeo que incluye: analizar sintácticamente un flujo de bits recibido para obtener información de predicción de una unidad de codificado (CU); cuando la CU tiene únicamente una unidad de transformada (TU) residual y un tamaño de la TU residual es menor que un tamaño de la CU, obtener un objetivo de modo de transformada de la TU residual; en donde el objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual y un conjunto de tipos de transformada de la TU residual; obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada; analizar sintácticamente el flujo de bits recibido para obtener coeficientes de transformada de la TU residual; aplicar una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados; aplicar, en función del tipo de transformada de la TU residual (o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual), una transformada inversa a los coeficientes descuantizados 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 sacar una secuencia de vídeo, la secuencia de vídeo incluye un fotograma de vídeo que incluye el bloque de vídeo.

Como el objetivo de modo de transformada puede especificar el modo de particionamiento de TU de la CU, la posición de la TU residual, y el tipo de transformada del residual, de modo que no únicamente se puede mejorar la eficiencia de decodificación, sino también se puede mejorar la calidad de decodificación; y para el objetivo de modo de

transformada se puede especificar la información anterior, así puede 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 audio codificados.

5 En una manera de implementación posible del primer aspecto, en donde N modos de transformada corresponden a un conjunto de tipos de transformada, en donde el conjunto de tipos de transformada incluye dos o más tipos de transformada o el conjunto de tipos de transformada incluye uno o más tipos de transformada. En una manera de implementación posible del primer aspecto, en donde el conjunto de tipos de transformada de cada modo de transformada incluye cuatro tipos de transformada, de la siguiente manera:

10 tipo de transformada 1: una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DST-7;
 tipo de transformada 2: una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-8, denotada como DST-7/DCT-8;
 15 tipo de transformada 3: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DCT-8/DST-7;
 tipo de transformada 4: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8, denotada como DCT-8/DCT-8.

20 En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otra manera de implementación posible del primer aspecto, en donde N modos de transformada corresponden a un conjunto de tipos de transformada, en donde el conjunto de tipos de transformada incluye dos o más tipos de transformada o el conjunto de tipos de transformada incluye uno o más tipos de transformada.

25 En una manera de implementación posible del primer aspecto, en donde el conjunto de tipos de transformada de cada modo de transformada incluye cuatro tipos de transformada, de la siguiente manera:

30 tipo de transformada 1: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4, denotada como DST-4/DST-4;
 tipo de transformada 2: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4, denotada como DST-4/DCT-4;
 tipo de transformada 3: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4, denotada como DCT-4/DST-4;
 35 tipo de transformada 4: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4, denotada como DCT-4/DCT-4.

En una manera de implementación posible del primer aspecto, en donde N modos de transformada corresponden a N conjuntos de tipos de transformada; en donde $N \geq 1$ y N es un entero positivo; en donde cada conjunto de tipos de transformada incluye dos o más tipos de transformada o cada conjunto de tipos de transformada incluye uno o más tipos de transformada.

45 En una manera de implementación posible del primer aspecto, si el conjunto de tipos de transformada incluye dos o más tipos de transformada, la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende:

seleccionar un tipo de transformada de la TU residual del conjunto de tipos de transformada.

50 En una manera de implementación posible del primer aspecto, el método comprende además:

analizar sintácticamente información de indicación del flujo de bits, en donde la información de indicación indica un tipo de transformada de la TU residual;
 la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende:
 55 determinar un tipo de transformada de la TU residual del conjunto de tipos de transformada en función de la información de indicación analizada sintácticamente del flujo de bits.

En una manera de implementación posible del primer aspecto, en donde si el conjunto de tipos de transformada de la TU residual incluye un primer tipo de transformada y un segundo tipo de transformada, el segundo tipo de transformada es dependiente del primer tipo de transformada; o
 60 si el conjunto de tipos de transformada de la TU residual incluye un primer tipo de transformada y un segundo tipo de transformada, el segundo tipo de transformada es independiente del primer tipo de transformada o el segundo tipo de transformada es un tipo de transformada unificado.

65 En una manera de implementación posible del primer aspecto, en donde el segundo tipo de transformada es un tipo de transformada unificado que es una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-2, denotada como DCT-2/DCT-2.

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

[illegible]

[illegible]

de la TU residual es inferior, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8; modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-2; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-1; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-1; modo de transformada VH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-8 o DCT-2; modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DCT-8 o DCT-2; o modo de transformada VH2 modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DST-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-8; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DST-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otra primera manera de implementación posible del primer aspecto, el tamaño de la CU se denota por $W \times H$; en donde el objetivo de modo de transformada de la TU residual es uno de los siguientes modos:

modo de transformada Q0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4; modo de transformada Q1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4; modo de transformada Q2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4; un segundo tipo de

[illegible]

modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4 o DCT-2; o modo de transformada VH2 modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DST-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-4; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DST-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DCT-4

Para otro ejemplo, el conjunto de tipos de transformada de cada modo de transformada puede contener cuatro tipos de transformada:

tipo 1: una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DST-7;
 tipo 2: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DCT-8;
 tipo 3: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DCT-8/DST-7;
 tipo 4: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8, denotada como DCT-8/DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, para otro ejemplo, el conjunto de tipos de transformada de cada modo de transformada puede contener cuatro tipos de transformada:

tipo 1: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4, denotada como DST-4/DST-4;
 tipo 2: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4, denotada como DST-4/DCT-4;
 tipo 3: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4, denotada como DCT-4/DST-4;
 tipo 4: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4, denotada como DCT-4/DCT-4.

El tipo de transformada de la TU residual se señala en flujo de bits. La información de indicación (tal como un índice o uno o más marcadores) se señala para indicar cuál de la pluralidad de tipos de transformada se usa para la TU residual. Cuando un conjunto de tipos de transformada contiene 2 tipos de transformada, se señala un marcador para indicar cuál de los 2 tipos de transformada se usa para la TU residual. Cuando un conjunto de tipos de transformada contiene 4 tipos de transformada, se señala un índice para indicar cuál de los 4 tipos de transformada se usa para la TU residual, y el índice puede codificarse por 2 bins o por código unario truncado. Por ejemplo, si la cantidad de tipos de transformada es más de dos, se pueden usar dos marcadores para indicar cuál de la pluralidad de tipos de transformada se usa para la TU residual, por ejemplo un marcador indica un tipo de transformada horizontal de la TU residual; y otro marcador indica un tipo de transformada vertical de la TU residual; en otras palabras, un primer marcador indica cuál de la pluralidad de tipos de transformada horizontales se usa para la TU residual y un segundo marcador indica cuál de la pluralidad de tipos de transformada verticales se usa para la TU residual.

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

El objetivo de modo de transformada puede ser indexado por el índice de modo, así la eficiencia de decodificación se puede mejorar para únicamente un índice necesario a obtener.

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

Determinar los objetivos de modos de transformada candidatos puede reducir la cantidad de objetivos de modos de transformada candidatos para la TU residual, por tanto reducir los bits necesarios para transmitir el índice de modo.

Con referencia a la tercera manera de implementación posible del primer aspecto, en una cuarta manera de implementación posible, en donde determinar objetivos de modos de transformada candidatos de la TU residual en función de un tamaño de la CU comprende al menos uno de los siguientes:

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VH0 y modo de transformada VH1;

cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HQ0, modo de transformada HQ1 y modo de transformada HH2;

cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HHO y modo de transformada HH1;

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VH0 y modo de transformada VH1;

cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HQ0, modo de transformada HQ1 y modo de transformada HH2; o

cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HHO y modo de transformada HH1;

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

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

Los modos de transformada se preagrupan, así puede reducir la cantidad de los modos de transformada adecuados para una CU, no únicamente puede mejorar la eficiencia de decodificación sino también puede ahorrar recurso de transmisión.

Con referencia a la quinta manera de implementación posible del primer aspecto, en una sexta manera de implementación posible, en donde al que pertenece el objetivo de modo de transformada cualquier grupo de modos según lo siguiente:

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

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

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

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

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

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

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

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

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

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

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

un grupo de modos comprende modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0 y modo de transformada VQ1.

Con referencia a la quinta manera de implementación posible del primer aspecto, en una séptima manera de implementación posible, en donde antes de analizar sintácticamente el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada, el método comprende además: determinar al menos un candidato de grupos de modos de la TU residual en función de un tamaño de la CU; en donde analizar sintácticamente el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada comprende: analizar sintácticamente, en función del al menos un candidato de grupos de modos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada.

Determinar los objetivos de modos de transformada candidatos puede reducir la cantidad de objetivos de modos de transformada candidatos para la TU residual, por tanto reducir los bits necesarios para transmitir el índice de modo.

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

cuando una anchura 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 modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

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

cuando una anchura 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 Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

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

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

Con referencia a la cuarta manera de implementación posible o la octava manera de implementación posible del primer aspecto, en una novena manera 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 manera de implementación posible del primer aspecto, en una décima manera de implementación posible, en donde antes de obtener el objetivo de modo de transformada de la TU residual, el método comprende además: determinar, en función de la información de predicción de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Con referencia a la décima manera de implementación posible del primer aspecto, en una decimoprimer manera de implementación posible, en donde determinar, en función de la información de predicción de la CU, si obtener el objetivo de modo de transformada de la TU residual comprende al menos uno de los siguientes:

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

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

cuando un método de predicción de la CU es un método de predicción positivo preestablecido, determinar obtener el objetivo de modo de transformada de la TU residual; o

cuando un método de predicción de la CU es un método de predicción negativo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual.

Con referencia a la decimoprimer manera de implementación posible del primer aspecto, en una decimosegunda manera de implementación posible, en donde el método de predicción positivo preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en modelo de traslación; un método de predicción por fusión; un método de predicción de vector de movimiento avanzado con precisión de diferencia de vector de movimiento 1/4-pel; o un método de predicción por fusión con un índice de fusión menor que 2.

Con referencia a la decimoprimer o decimosegunda manera de implementación posible del primer aspecto, en una decimotercera manera de implementación posible, en donde el método de predicción negativo preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en modelo afin; un afin método de predicción por fusión; un modo de interpredicción afin; un método de predicción de vector de movimiento avanzado con precisión de diferencia de vector de movimiento 1-pel o 4-pel; o un método de predicción por fusión con un índice de fusión no menor que 2.

Con referencia al primer aspecto o una cualquiera de las maneras de implementación posibles primera a decimotercera del primer aspecto, en una decimocuarta manera de implementación posible, en donde antes de obtener el objetivo de modo de transformada de la TU residual, el método comprende además: determinar, en función de un tamaño de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Con referencia a la decimocuarta manera de implementación posible del primer aspecto, en una decimoquinta manera de implementación posible, determinar, en función de un tamaño de la CU, si obtener el objetivo de modo de transformada de la TU residual comprende al menos uno de los siguientes:

cuando una anchura o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual;

cuando una anchura o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual; o

cuando una anchura o una altura de una TU de la CU es mayor que un umbral de TU máximo prestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual.

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

Según un segundo aspecto, se proporciona un decodificador de vídeo, que comprende: una unidad de decodificación por entropía, se configura para analizar sintácticamente un flujo de bits recibido para obtener información de predicción de una unidad de codificado (CU); obtener un objetivo de modo de transformada de una unidad de transformada residual (TU) 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 objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual, y un conjunto de tipos de transformada de la TU residual; y analizar sintácticamente el flujo de bits recibido para obtener coeficientes de transformada de la TU residual; en donde la unidad de decodificación por entropía se configura además para obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada; una unidad de procesamiento de cuantización inversa, se configura para aplicar una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados; una unidad de procesamiento de transformada inversa, se configura para aplicar, en función del tipo de transformada de la TU residual o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual, una transformada inversa a los coeficientes descuantizados para obtener un bloque residual de la TU residual; una unidad de procesamiento de predicción, se configura para obtener un bloque de predicción de la CU en función de la información de predicción; una unidad de reconstrucción, se configura para obtener un bloque de vídeo en función del bloque residual y el bloque de predicción; y un salida, se configura para tener como salida una secuencia de vídeo, la secuencia de vídeo incluye un fotograma de vídeo que incluye el bloque de vídeo.

En una manera de implementación posible del segundo aspecto, en donde N modos de transformada corresponden a un conjunto de tipos de transformada, en donde el conjunto de tipos de transformada incluye dos o más tipos de transformada o el conjunto de tipos de transformada incluye uno o más tipos de transformada.

En una manera de implementación posible del segundo aspecto, en donde el conjunto de tipos de transformada de cada modo de transformada incluye cuatro tipos de transformada, de la siguiente manera:

tipo de transformada 1: una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DST-7;
 tipo de transformada 2: una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-8, denotada como DST-7/DCT-8;
 tipo de transformada 3: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DCT-8/DST-7;
 tipo de transformada 4: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8, denotada como DCT-8/DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en una manera de implementación posible del segundo aspecto, en donde el conjunto de tipos de transformada de cada modo de transformada incluye cuatro tipos de transformada, de la siguiente manera:

tipo de transformada 1: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4, denotada como DST-4/DST-4;
 tipo de transformada 2: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4, denotada como DST-4/DCT-4;
 tipo de transformada 3: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4, denotada como DCT-4/DST-4;
 tipo de transformada 4: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4, denotada como DCT-4/DCT-4.

En una manera de implementación posible del segundo aspecto, en donde N modos de transformada corresponden a N conjuntos de tipos de transformada; en donde $N \geq 1$ y N es un entero positivo; en donde cada conjunto de tipos de transformada incluye dos o más tipos de transformada o cada conjunto de tipos de transformada incluye uno o más tipos de transformada.

En una manera de implementación posible del segundo aspecto, si el conjunto de tipos de transformada incluye dos o más tipos de transformada, la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende:

seleccionar un tipo de transformada de la TU residual del conjunto de tipos de transformada.

En una manera de implementación posible del segundo aspecto, el método comprende además: analizar sintácticamente información de indicación del flujo de bits, en donde la información de indicación indica un tipo de transformada de la TU residual;

la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende: determinar un tipo de transformada de la TU residual del conjunto de tipos de transformada en función de la información de indicación analizada sintácticamente del flujo de bits.

En una manera de implementación posible del segundo aspecto, en donde si el conjunto de tipos de transformada de la TU residual incluye un primer tipo de transformada y un segundo tipo de transformada, el segundo tipo de transformada es dependiente del primer tipo de transformada; o

si el conjunto de tipos de transformada de la TU residual incluye un primer tipo de transformada y un segundo tipo de transformada, el segundo tipo de transformada es independiente del primer tipo de transformada o el segundo tipo de transformada es un tipo de transformada unificado.

En una manera de implementación posible del segundo aspecto, en donde el segundo tipo de transformada es un tipo de transformada unificado que es una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-2, denotada como DCT-2/DCT-2.

En una primera manera de implementación posible del segundo aspecto, el tamaño de la CU se denota por $W \times H$; en donde el objetivo de modo de transformada de la TU residual es uno de los siguientes modos:

modo de transformada Q0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7;

modo de transformada Q1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DST-7;

modo de transformada Q2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-8;

modo de transformada Q3, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DST-8;

modo de transformada QH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 3 TU, en donde 2 TU de la 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7;

modo de transformada QH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 3 TU, en donde 2 TU de la 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DST-7;

de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-8 o DCT-2;

modo de transformada VQ1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8 o DCT-2;

modo de transformada HH0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-7;

modo de transformada HH1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8;

modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada; en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-2; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-1; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-1;

modo de transformada VH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2;

modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DCT-8 o DCT-2; o

modo de transformada VH2 modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DST-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-8; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DST-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en una primera manera de implementación posible del segundo aspecto, el tamaño de la CU se denota por $W \times H$; en donde el objetivo de modo de transformada de la TU residual es uno de los siguientes modos:

vertical de la TU residual es DCT-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 o DCT-2, y una transformada vertical de la TU residual es DST-4;

modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 centro, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-2 un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-2; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-1; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-1;

modo de transformada VH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4 o DCT-2;

modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4 o DCT-2; o

modo de transformada VH2 modo de transformada HH2, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DST-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-2 y una transformada vertical de la TU residual es DCT-4; o en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DST-4; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-1 y una transformada vertical de la TU residual es DCT-4

Para otro ejemplo, el conjunto de tipos de transformada de cada modo de transformada puede contener cuatro tipos de transformada:

tipo 1: una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DST-7;

tipo 2: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DCT-8;

tipo 3: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DCT-8/DST-7;

tipo 4: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8, denotada como DCT-8/DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, para otro ejemplo, el conjunto de tipos de transformada de cada modo de transformada puede contener cuatro tipos de transformada:

tipo 1: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4, denotada como DST-4/DST-4;

tipo 2: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4, denotada como DST-4/DCT-4;

tipo 3: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4, denotada como DCT-4/DST-4;

tipo 4: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4, denotada como DCT-4/DCT-4.

Con referencia al primer aspecto o la primera manera de implementación posible del segundo aspecto, en una segunda manera de implementación posible, en donde la unidad de decodificación por entropía se configura además para

analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada de la TU residual; y obtener el objetivo de modo de transformada de la TU residual en función del índice de modo.

5 Con referencia a la segunda manera de implementación posible del segundo aspecto, en una tercera manera de implementación posible, en donde la unidad de decodificación por entropía, se configura además para determinar objetivos de modos de transformada candidatos de la TU residual en función de un tamaño de la CU; y analizar sintácticamente, en función de los objetivos de modos de transformada candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar los objetivos de modo de transformada de la TU residual.

10 Con referencia a la tercera manera de implementación posible del segundo aspecto, en una cuarta manera de implementación posible, en donde la unidad de decodificación por entropía se configura además para determinar objetivos de modos de transformada candidatos de la TU residual según al menos uno de los siguientes:

15 cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;
cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VQ0, modo
20 de transformada VQ1 y modo de transformada VH2;
cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VH0 y modo de transformada VH1;
cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HQ0, modo
25 de transformada HQ1 y modo de transformada HH2;
cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HHO y modo de transformada HH1;
cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;
cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VQ0, modo
35 de transformada VQ1 y modo de transformada VH2;
cuando una anchura 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 objetivos de modos de transformada candidatos comprenden modo de transformada VH0 y modo de transformada VH1;
cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HQ0, modo de transformada HQ1 y modo de transformada HH2; o
cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos comprenden modo de transformada HHO y modo de transformada HH1;
40 en donde Th1, Th2 y Th3 son valores enteros predefinidos.

Con referencia al segundo aspecto o la primera manera de implementación posible del segundo aspecto, en una quinta manera de implementación posible, en donde la unidad de decodificación por entropía se configura además para analizar sintácticamente el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada; y analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada en el grupo de modos; y obtener el objetivo de modo de transformada en función del índice de grupo y el índice de modo.

55 Con referencia a la quinta manera de implementación posible del segundo aspecto, en una sexta manera de implementación posible, en donde al que pertenece el objetivo de modo de transformada cualquier grupo de modos según lo siguiente:

un grupo de modos comprende modo de transformada HH0, modo de transformada HH1, modo de transformada HH2, modo de transformada VH0, modo de transformada VH1 y modo de transformada VH2;
60 un grupo de modos comprende modo de transformada Q0, modo de transformada Q1, modo de transformada Q2, modo de transformada Q3, modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0 y modo de transformada VQ1;
un grupo de modos comprende modo de transformada HH0, HH1, HH2, VH0, VH1 y VH2;
un grupo de modos comprende modo de transformada Q0, modo de transformada Q1, modo de transformada Q2, modo de transformada Q3, modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0 y modo de transformada VQ1;
65

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

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

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

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

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

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

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

un grupo de modos comprende modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0 y modo de transformada VQ1.

Con referencia a la quinta manera de implementación posible del segundo aspecto, en una séptima manera de implementación posible, en donde la unidad de decodificación por entropía se configura además para determinar al menos un candidato de grupos de modos de la TU residual en función de un tamaño de la CU; y analizar sintácticamente, en función del al menos un candidato de grupos de modos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada.

Con referencia a la séptima manera de implementación posible del segundo aspecto, en una octava manera de implementación posible, en donde la unidad de decodificación por entropía se configura además para determinar al menos un candidato de grupos de modos según al menos uno de los siguientes:

cuando una anchura 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 modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

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

cuando una anchura 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 Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los grupos de modos candidatos comprenden un grupo de modos que comprende modo de transformada HHO y modo de transformada HH1;
en donde Th1, Th2 y Th3 son valores enteros predefinidos.

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

10 Con referencia al segundo aspecto o una cualquiera de las maneras de implementación posibles primera a novena del segundo aspecto, en una décima manera de implementación posible, en donde la unidad de decodificación por entropía, se configura además para determinar, en función de la información de predicción de la CU, si obtener el objetivo de modo de transformada de la TU residual.

15 Con referencia a la décima manera de implementación posible del segundo aspecto, en una decimoprimer manera de implementación posible, en donde la unidad de decodificación por entropía, se configura además para determinar si obtener el objetivo de modo de transformada de la TU residual según al menos uno de los siguientes:

20 cuando un modo de predicción de la CU es modo de intrapredicción, determinar obtener el objetivo de modo de transformada de la TU residual;
cuando un modo de predicción de la CU es modo de interpretación, determinar obtener el objetivo de modo de transformada de la TU residual;
cuando un método de predicción de la CU es un método de predicción positivo preestablecido, determinar obtener el objetivo de modo de transformada de la TU residual; o
25 cuando un método de predicción de la CU es un método de predicción negativo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual.

30 Con referencia a la decimoprimer manera de implementación posible del segundo aspecto, en una decimosegunda manera de implementación posible, en donde el método de predicción positivo preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en modelo de traslación; un método de predicción por fusión; un método de predicción de vector de movimiento avanzado con precisión de diferencia de vector de movimiento 1/4-pel; o un método de predicción por fusión con un índice de fusión menor que 2.

35 Con referencia a la decimoprimer o decimosegunda manera de implementación posible del segundo aspecto, en una decimotercera manera de implementación posible, en donde el método de predicción negativo preestablecido comprende al menos uno de los siguientes: un método de compensación de movimiento basado en modelo afin; un afin método de predicción por fusión; un modo de interpretación afin; un método de predicción de vector de movimiento avanzado con precisión de diferencia de vector de movimiento 1-pel o 4-pel; o un método de predicción por fusión con un índice de fusión no menor que 2.

40 Con referencia al segundo aspecto o una cualquiera de la primera a decimotercera manera de implementación posible del segundo aspecto, en una decimocuarta manera de implementación posible, en donde la unidad de decodificación por entropía, se configura además para determinar, en función de un tamaño de la CU, si obtener el objetivo de modo de transformada de la TU residual.

45 Con referencia a la decimocuarta manera de implementación posible del segundo aspecto, en una decimoquinta manera de implementación posible, la unidad de decodificación por entropía, se configura además para determinar si obtener el objetivo de modo de transformada de la TU residual según al menos uno de los siguientes:

50 cuando una anchura o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual;
cuando una anchura o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual; o
cuando una anchura o una altura de una TU de la CU es mayor que un umbral de TU máximo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual.

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

60 Según un tercer aspecto, se proporciona un decodificador que comprende circuitería de procesamiento para llevar a cabo el método según uno cualquiera del primer aspecto y las maneras de implementación imposibles primera a decimosexta del primer aspecto.

65

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 maneras de implementación imposibles primera a decimosexta del primer aspecto.

5 Según un quinto aspecto, se proporciona un decodificador que comprende: uno o más procesadores; y un soporte de almacenamiento no transitorio legible por ordenador acoplado a los procesadores y que almacenan programación para ejecución 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 maneras de implementación imposibles primera a decimosexta del primer aspecto.

10 Al realizar métodos proporcionados por la presente divulgación, el objetivo de modo de transformada puede especificar el modo de particionamiento de TU de la CU, la posición de la TU residual, y el tipo de transformada del residual, de modo que no únicamente se puede mejorar la eficiencia de decodificación, sino también se puede mejorar la calidad de decodificación; y para el objetivo de modo de transformada se puede especificar la información anterior, así puede
15 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 audio codificados.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

20 Para describir más claramente las soluciones técnicas en las realizaciones de la presente invención o en la técnica anterior, 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 meramente algunas realizaciones de la presente invención, y un experto en la técnica todavía puede derivar otros dibujos adjuntos a partir de estos dibujos sin esfuerzos creativos.

25 La Figura 1A es un diagrama de bloques esquemático o conceptual que ilustra un ejemplo de sistema de codificado.
La Figura 1B es un diagrama ilustrativo de otro ejemplo de sistema de codificado de vídeo.
La Figura 2 es un diagrama de bloques esquemático/conceptual de un ejemplo de un codificador de vídeo.
30 La Figura 3 es un diagrama de bloques esquemático/conceptual de un ejemplo de un decodificador de vídeo.
La Figura 4 es un diagrama esquemático de un dispositivo de codificado de vídeo.
La Figura 5 es un diagrama de bloques de un aparato.
La Figura 6 ilustra modos de transformada Q0, Q1, Q2 y Q3 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
35 La Figura 7 ilustra modos de transformada HQ0, HQ1, VQ0 y VQ1.
La Figura 8 ilustra modos de transformada HH0, HH1, VH0 y VH1.
La Figura 9 ilustra modos de transformada HH2 y VH2 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
La Figura 10 ilustra modos de transformada QH0, QH1, QH2 y QH3 como ejemplo que no comprende todos
40 los rasgos necesarios para implementar la presente invención.
La Figura 11 ilustra modos de transformada QV0, QV1, QV2 y QV3 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
La Figura 12A y la Figura 12B son diagramas esquemáticos de un método de decodificación de vídeo.
La Figura 13 ilustra un ejemplo de intrapredicción en codificado de vídeo.
45 La FIGURA 14 es un diagrama esquemático de un ejemplo de un mecanismo de codificación de vídeo.
Las FIGURAS 15-16 ilustran ejemplos de transformadas de transformada variable espacial (SVT).
La Figura 17 ilustra ejemplos de posiciones candidatas de transformada SVT respecto a un bloque residual.
La Figura 18 ilustra ejemplos de posiciones de transformada SVT respecto a un bloque residual.
La Figura 19 ilustra un ejemplo de volteo horizontal de muestra residual.
50 La Figura 20 es un diagrama de flujo de un ejemplo de método de decodificación de vídeo con SVT dependiente de posición.
La Figura 21 es una ilustración de posiciones candidatas de bloques SVT-V y SVT-H ($M1=M2=8$) como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
La Figura 22 es una ilustración de SVT-V y SVT-H con 3 posiciones candidatas como ejemplo que no
55 comprende todos los rasgos necesarios para implementar la presente invención.
La Figura 23 es una ilustración de SVT-H y SVT-V como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
La Figura 24 es una ilustración de posiciones candidatas de bloques SVT-V y SVT-H ($M1=M2=8$) como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.
60 La Figura 25 es una ilustración de un primer grupo de coeficientes de transformada y un segundo grupo de coeficientes de transformada.

DESCRIPCIÓN DE REALIZACIONES

65 En la siguiente descripción, se hace referencia a las figuras adjuntas, que forman parte de esta divulgación, y que muestran, a modo de ilustración, aspectos específicos de las realizaciones de la invención o aspectos específicos en

los que se pueden usar las realizaciones de la presente invención. Debe entenderse que las realizaciones de la invención se pueden usar en otros aspectos, y comprende cambios estructurales o lógicos no representados en las figuras. La siguiente descripción detallada, por lo tanto, no se debe tomar en un sentido limitativo, y la presente invención está definida por las reivindicaciones anexas.

Por ejemplo, se entiende que una divulgación en relación con un método descrito también siguen siendo verdadera para un correspondiente dispositivo o sistema configurado para realizar el método y viceversa. Por ejemplo, si se describe una o una pluralidad de etapas de método específicas, un dispositivo correspondiente puede incluir una o una pluralidad de unidades, por ejemplo unidades funcionales, para realizar la una o más etapas de método descritas (por ejemplo, una unidad que realiza la una o una pluralidad de etapas; o una pluralidad de unidades cada una realizando una o más de la pluralidad de etapas), incluso si tal una o más unidades no se describen o ilustran explícitamente en las figuras. Por otro lado, por ejemplo, si se describe un aparato específico basado en una o una pluralidad de unidades, por ejemplo unidades funcionales, un método correspondiente puede incluir una etapa usada para realizar la funcionalidad de la una o una pluralidad de unidades (por ejemplo, una etapa usada para realizar la funcionalidad de la una o una pluralidad de unidades, o una pluralidad de etapas que realizan cada una 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 los diversos aspectos y/o realizaciones ejemplares descritos en esta memoria pueden combinarse entre sí, a menos que se indique específicamente de otro modo.

Codificado de vídeo típicamente se refiere al procesamiento de una secuencia de imágenes, que forman el vídeo o la secuencia de vídeo. En lugar del término "ilustración" se puede usar el término "fotograma" o "imagen" como sinónimos en el campo de codificado de vídeo. El codificado de vídeo usado en la presente solicitud (o presente divulgación) indica tanto codificación de vídeo como decodificación de vídeo. El codificado de vídeo se realiza en un lado de fuente, y típicamente comprende procesar (por ejemplo, por compresión) las imágenes de vídeo original para reducir una cantidad de datos requeridos para representar las imágenes de vídeo (para almacenamiento y/o transmisión más eficientes). La decodificación de vídeo se realiza en el lado de destino y típicamente comprende el procesamiento inverso comparado con el codificador para reconstruir las imágenes de vídeo. Realizaciones que hacen referencia a "codificado" de imágenes de vídeo (o imágenes en general, como se explicarán más tarde) se entenderán relacionadas con "codificación" o "decodificación" para secuencia de vídeo. La combinación de una parte de codificación y una parte de decodificación también se denomina CODEC (codificación y decodificación).

En caso de codificado de vídeo sin pérdida, las imágenes de vídeo originales se pueden reconstruir, es decir, las imágenes de vídeo reconstruidas tienen la misma calidad que las imágenes de vídeo originales (asumiendo ausencia de pérdida de transmisión u otra pérdida de datos durante el almacenamiento o la transmisión). En caso de codificado de vídeo con pérdidas, se realiza una compresión adicional, por ejemplo mediante cuantización, 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 codificado 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 codificado de transformada 2D para aplicar cuantización en el dominio de transformada). Cada imagen de una secuencia de vídeo típicamente se particiona en un conjunto de bloques no superpuestos, y típicamente el codificado se realiza a nivel de bloque. En otras palabras, en el codificador típicamente se procesa el vídeo, es decir, se codifica, a nivel de bloque (bloque de vídeo), por ejemplo usando predicción espacial (intraimagen) y predicción temporal (interimagen) para generar un bloque de predicción, sustraer el bloque de predicción del bloque actual (bloque actualmente procesado/a procesar) para obtener un bloque residual, transformar el bloque residual y quantizar el bloque residual en el dominio de transformada para reducir la cantidad de datos a transmitir (compresión), mientras que en el decodificador el procesamiento inverso comparado con el codificador se aplica parcialmente al bloque codificado o comprimido para reconstruir el bloque actual para representación. Es más, el codificador duplica el bucle de procesamiento de decodificador de manera que ambos generarán predicciones idénticas (por ejemplo intra- y interpredicciones) y/o re-construcciones para procesar, es decir, codificar, los subsiguientes bloques.

Como se emplea en esta memoria, el término "bloque" puede ser parte de una imagen o un fotograma. Para conveniencia de la descripción, en esta memoria se describen realizaciones de la invención en referencia a Codificado de Vídeo de Alta Eficiencia (HEVC) o el software de referencia de codificado de vídeo Versátil (VVC), desarrollado por Joint Collaboration Team en Video Coding (JCT-VC) del Grupo de Expertos de Codificado de Vídeo ITU-T (VCEG) y el Grupo de Expertos de Imágenes en Movimiento ISO/IEC (MPEG). Un experto en la técnica entenderá que las realizaciones de la invención no se limitan a HEVC o VVC. Puede referirse a una CU, una PU y una TU. En HEVC, una CTU se parte en CU usando una estructura de árbol cuádruple denotado como árbol de codificado. La decisión sobre si codificar un área de imagen usando predicción interimagen (temporal) o intraimagen (espacial) se hace a nivel de CU. Cada CU se puede partir además en una, dos o cuatro PU según el tipo de partición de PU. Dentro de una PU, se aplica el mismo proceso de predicción, y la información relevante se transmite al decodificador en una base de PU. Tras obtener el bloque residual al aplicar el proceso de predicción basado en el tipo de partición de PU, una CU puede particionarse en unidades de transformada (TU) según otra estructura de árbol cuádruple similar al árbol de

codificado para la CU. En el desarrollo más nuevo de la técnica de compresión de vídeo, un marco de partición de árbol cuádruple y árbol binario (QTBT) se usa para particionar un bloque de codificado. En la estructura de bloques QTBT, una CU puede tener una forma cuadrada o rectangular. Por ejemplo, una unidad de árbol de codificado (CTU) se particiona primero por una estructura de árbol cuádruple. Los nodos de hoja de árbol cuádruple se particionan además por una estructura de árbol binario. Los nodos de hoja de árbol binario se denominan unidades de codificado (CU), y esa segmentación se usa para procesamiento por predicción y transformada sin particionado adicional. Esto significa que CU, PU y TU tienen el mismo tamaño de bloque en la estructura de bloques de codificado QTBT. En paralelo, también se ha propuesto usar partición por multiplicación, por ejemplo, partición por triple árbol junto con la estructura de bloques QTBT.

En las siguientes realizaciones de un codificador 20, un decodificador 30 y un sistema de codificado 10 se describen en función de las Figuras 1 a 3.

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

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

La fuente de imágenes 16 puede comprender o ser cualquier clase de dispositivo de captura de imágenes, por ejemplo para capturar una imagen del mundo real, y/o cualquier clase de dispositivo de generación de imagen o comentario (para codificado de contenido en pantalla, algunos textos en la pantalla también se consideran parte de una ilustración o imagen a codificar), por ejemplo un procesador de gráficos por ordenador para generar una imagen animada por ordenador, o cualquier clase de dispositivo para obtener y/o proporcionar una imagen del mundo real, una imagen animada por ordenador (por ejemplo un contenido en pantalla, una imagen de realidad virtual (VR)) y/o cualquier combinación de los mismos (por ejemplo una imagen de realidad aumentada (AR)).

Una imagen (digital) es o se puede considerar como distribución o matriz bidimensional de muestras con valores de intensidad. Una muestra en la distribución también se pueden denominar píxel (forma corto de elemento de imagen) o un pel. El número de muestras en dirección (o eje) horizontal y vertical de la distribución o la imagen define el tamaño y/o la resolución de la imagen. Para representación de color, típicamente se emplean tres componentes de color, es decir, la imagen puede representarse o incluir tres distribuciones de muestras. En formato RBG o espacio de color, una imagen comprende una distribución de muestras correspondientes de rojo, verde y azul. Sin embargo, en codificado de vídeo cada píxel típicamente se representa en un formato de luminancia/crominancia o espacio de color, por ejemplo YCbCr, que comprende un componente de luminancia indicado por Y (a veces en cambio también se usa L) y dos componentes de crominancia indicados por Cb y Cr. El componente de luminancia (o corto luma) Y representa la intensidad de nivel de gris o brillo (por ejemplo como en una imagen en escala de grises), mientras los dos componentes de crominancia (o croma por abreviar) Cb y Cr representan los componentes de información de cromaticidad o color. Por consiguiente, la imagen en formato YCbCr comprende una distribución de muestras de luminancia de valores de muestra de luminancia (Y) y dos distribuciones de muestras de crominancia de valores de crominancia (Cb y Cr). Imágenes en formato RGB pueden convertirse o transformarse 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 únicamente una distribución de muestras de luminancia.

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

En distinción a la unidad de preprocesamiento 18 y el procesamiento realizado por la unidad de preprocesamiento 18, la imagen o los datos de imagen 17 (por ejemplo datos de vídeo 16) también se pueden denominar imagen sin procesar o datos de imagen sin procesar 17.

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

El decodificador 20 (por ejemplo codificador de vídeo 20) se configura para recibir los datos de imagen preprocesados 19 y proporcionar datos de imagen codificados 21 (detalles adicionales se describirán más adelante, por ejemplo sobre la base de la Figura 2 o la Figura 4).

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

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

La interfaz de comunicación 28 del dispositivo de destino 14 se configura recibir los datos de imagen codificados 21 o los datos codificados 13, por ejemplo directamente del dispositivo fuente 12 o de cualquier otra fuente, por ejemplo un dispositivo de almacenamiento, por ejemplo un dispositivo de almacenamiento de datos de imagen codificados.

La interfaz de comunicación 22 y la interfaz de comunicación 28 se pueden configurar para transmitir o recibir los datos de imagen codificados 21 o datos codificados 13 por medio de un enlace de comunicación directa entre el dispositivo fuente 12 y el dispositivo de destino 14, por ejemplo una conexión cableado directa o inalámbrica, o por medio de cualquier clase de red, por ejemplo una red cableada o inalámbrica o cualquier combinación de las mismas, o cualquier clase de red privada y pública, o cualquier clase de combinación de los mismos.

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

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

Ambas, interfaz de comunicación 22 e interfaz de comunicación 28 se pueden configurar como interfaces de comunicación unidireccional como se indica por la flecha para los datos de imagen codificados 13 en la Figura 1A apuntando desde el dispositivo fuente 12 al dispositivo de destino 14, o interfaces de comunicación bidireccional, y se puede configurar, por ejemplo para enviar y recibir mensajes, por ejemplo para configurar una conexión, para acusar recibo e intercambiar cualquier otra información relacionada con el enlace de comunicación y/o transmisión de datos, por ejemplo transmisión de datos de imagen codificados.

El decodificador 30 se configura para recibir los datos de imagen codificados 21 y proporcionar datos de imagen decodificados 31 o una imagen decodificada 31 (detalles adicionales se describirán más adelante, por ejemplo sobre la base de la Figura 3 o la Figura 5).

El posprocesador 32 del dispositivo de destino 14 se configura para posprocesar los datos de imagen de codificados 31 (también llamados datos de imagen reconstruidos), por ejemplo la imagen decodificada 31, para obtener datos de imagen posprocesados 33, por ejemplo una imagen posprocesada 33. El posprocesamiento realizado por la unidad de posprocesamiento 32 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 preparar los datos de imagen de decodificados 31 para exponer, por ejemplo por el dispositivo de exposición 34.

El dispositivo de exposición 34 del dispositivo de destino 14 se configura para recibir los datos de imagen posprocesados 33 para exponer la imagen, por ejemplo a un usuario o espectador. El dispositivo de exposición 34 puede ser o puede comprender cualquier tipo de pantalla para presentar una imagen reconstruida, por ejemplo, una pantalla o monitor integrados o externos. Las pantallas pueden, por ejemplo comprender pantallas de cristal líquido (LCD), pantallas de diodos emisores de luz orgánicos (OLED), pantallas de plasma, un proyector, pantallas de microleds, un cristal líquido en silicio (LCoS), un procesador de luz digital (DLP), o cualquier otro tipo de pantalla.

Aunque la Figura 1A representa el dispositivo fuente 12 y el dispositivo de destino 14 como dispositivos separados, realizaciones de dispositivos también pueden comprender ambos o ambas funcionalidades, el dispositivo fuente 12 o la funcionalidad correspondiente y el dispositivo de destino 14 o la funcionalidad correspondiente. En este tipo de

realizaciones, el dispositivo fuente 12 o la funcionalidad correspondiente y el dispositivo de destino 14 o la funcionalidad correspondiente pueden implementarse usando el mismo hardware y/o software, o hardware y/o software separados, o cualquier combinación de los mismos.

- 5 Como será evidente para un experto en la técnica sobre la base de la descripción, la existencia y la partición (exacta) de funcionalidades de diferentes unidades o funcionalidades del dispositivo fuente 12 y/o el dispositivo de destino 14 como se muestran en la Figura 1A pueden variar dependiendo de un dispositivo real y la aplicación.

10 El codificador 20 (por ejemplo un codificador de vídeo 20) y el decodificador 30 (por ejemplo un decodificador de vídeo 30) pueden implementarse cada uno como cualquiera de una variedad de circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados de aplicación específica (ASIC), matrices de puertas programables en campo (FPGA), lógica discreta, hardware, o cualquier combinación de los mismos. Si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un soporte de almacenamiento legible por ordenador no transitorio adecuado y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las tecnologías de esta divulgación. 15 Cualquiera de los anteriores (incluido hardware, software, una combinación de hardware y software, etc.) puede considerarse que es uno o más procesadores. Cada uno de codificador de vídeo 20 y el decodificador de vídeo 30 se puede incluir en uno o más codificadores o decodificadores, cualquiera de ellos puede integrarse como parte de un codificador/decodificador combinado (CÓDEC) en un dispositivo respectivo.

20 El dispositivo fuente 12 puede denominarse un dispositivo de codificación de vídeo o un aparato de codificación de vídeo. El dispositivo de destino 14 puede denominarse un dispositivo de decodificación de vídeo o un aparato de decodificación vídeo. El dispositivo fuente 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificado de vídeo o aparatos de codificado de vídeo.

25 El dispositivo fuente 12 y el dispositivo de destino 14 pueden comprender cualquiera de un amplio abanico de dispositivos, incluida cualquier clase de dispositivos de mano o estacionarios, por ejemplo ordenador notebook u ordenadores portátiles, teléfonos móviles, teléfonos inteligentes, tabletas u ordenadores tableta, cámaras, ordenadores de escritorio, descodificadores, televisiones, el dispositivos de exposición, reproductores multimedia, consolas de videojuegos, el dispositivos de retransmisión de vídeo retransmisión (tales como servidores de servicios de contenido o servidores de entrega de contenido), el dispositivo receptor de difusión, el dispositivo transmisor de difusión, o algo semejante y puede usar ninguno o cualquier clase de sistema operativo.

30 En algunos casos, el dispositivo fuente 12 y el dispositivo de destino 14 pueden equiparse para comunicación inalámbrica. Así, el dispositivo fuente 12 y el dispositivo de destino 14 pueden ser dispositivos de comunicación inalámbrica.

35 En algunos casos, el sistema de codificado de vídeo 10 ilustrado en la FIG 1A es meramente un ejemplo y las técnicas de la presente solicitud pueden aplicarse a ajustes de codificado de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no necesariamente incluyen cualquier comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, pueden recuperarse datos de una memoria local, transmitirse por una red, o algo semejante. Un dispositivo de codificación de vídeo puede codificar y almacenar datos en una memoria y/o un dispositivo de decodificación de vídeo puede recuperar y decodificar datos de la memoria. En algunos ejemplos, codificación y decodificación se realizan por dispositivos que no se comunican entre sí, sino que simplemente codifican 40 datos en una memoria y/o recuperan y decodifican datos de la memoria.

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

50 La Figura 1B es un diagrama ilustrativo de otro ejemplo de sistema de codificado de vídeo 40 que incluye el codificador 20 de la figura 2 y/o el decodificador 30 de la figura 3 según una realización ejemplar. El sistema 40 puede implementar técnicas según diversos ejemplos descritos en la presente solicitud. En la implementación ilustrada, el sistema de codificado de vídeo 40 puede incluir dispositivo(s) de formación de imágenes 41, el codificador de vídeo 100, el decodificador de vídeo 30 (y/o un codificador de vídeo implementado por medio de circuitería lógica 47 de unidad(es) 55 de procesamiento 46), una antena 42, uno o más procesador(es) 43, uno o más almacenamiento(s) en memoria 44, y/o un dispositivo de exposición 45.

60 Como se ilustra, dispositivo(s) de formación de imágenes 41, antena 42, unidad(es) de procesamiento 46, circuitería lógica 47, codificador de vídeo 20, decodificador de vídeo 30, procesador(es) 43, almacenamiento(s) en memoria 44 y/o dispositivo de exposición 45 pueden ser capaces de comunicación entre sí. Como se discute, aunque se ilustra 65

con ambos codificador de vídeo 20 y decodificador de vídeo 30, el sistema de codificado de vídeo 40 puede incluir únicamente codificador de vídeo 20 o únicamente decodificador de vídeo 30 en diversos ejemplos.

Como se muestra, en algunos ejemplos, el sistema de codificado de vídeo 40 puede incluir antena 42. La antena 42 se puede configurar para transmitir o recibir un flujo de bits codificado de datos de vídeo, por ejemplo. Además, en algunos ejemplos, el sistema de codificado de vídeo 40 puede incluir dispositivo de exposición 45. El dispositivo de exposición 45 se puede configurar para presentar datos de vídeo. Como se muestra, en algunos ejemplos, la circuitería lógica 47 puede implementarse por medio de unidad(es) de procesamiento 46. La(s) unidad(es) de procesamiento 46 pueden incluir lógico de circuito integrado de aplicación específica (ASIC), procesador(es) de gráficos, procesador(es) de finalidad general, o algo semejante. El sistema de codificado de vídeo 40 también puede incluir procesador(es) opcional(es) 43, que de manera similar pueden incluir lógica de circuito integrado de aplicación específica (ASIC), procesador(es) de gráficos, procesador(es) de finalidad general, o algo semejante. En algunos ejemplos, la circuitería lógica 47 puede implementarse por medio de hardware, hardware dedicado para codificado de vídeo, o algo semejante, y el (los) procesador(es) 43 pueden implementar software de finalidad general, sistemas operativos, o algo semejante. Adicionalmente, almacenamiento(s) en memoria 44 puede ser cualquier tipo de memoria tal como memoria volátil (por ejemplo, Memoria de Acceso Aleatorio Estática (SRAM), Memoria de Acceso Aleatorio Dinámico (DRAM), etc.) o memoria no volátil (por ejemplo, memoria flash, etc.), etc.. En un ejemplo no limitativo, pueden implementarse almacenamiento(s) en memoria 44 por memoria caché. En algunos ejemplos, la circuitería lógica 47 puede acceder a almacenamiento(s) en memoria 44 (para implementación de un búfer de imágenes por ejemplo). En otros ejemplos, la circuitería lógica 47 y/o las unidad(es) de procesamiento 46 pueden incluir almacenes de memoria (por ejemplo, caché o algo semejante) para la implementación de un búfer de imágenes o algo semejante.

En algunos ejemplos, el codificador de vídeo 100 implementado por medio de circuitería lógica puede incluir un búfer de imágenes (por ejemplo, por medio de unidad(es) de procesamiento 46 o almacenamiento(s) en memoria 44)) y una unidad de procesamiento de gráficos (por ejemplo, por medio de unidad(es) de procesamiento 46). La unidad de procesamiento de gráficos puede acoplarse comunicativamente al búfer de imágenes. La unidad de procesamiento de gráficos puede incluir codificador de vídeo 100 como implementado por medio de circuitería lógica 47 para materializar los diversos módulos que se discuten con respecto a la FIG 2 y/o cualquier otro sistema o subsistema de codificador descrito en esta memoria. La circuitería lógica se puede configurar para realizar las diversas operaciones como se discute en esta memoria.

El decodificador de vídeo 30 puede implementarse de manera similar como implementado por medio de circuitería lógica 47 para materializar los diversos módulos como se discute con respecto al decodificador 30 de la FIG 3 y/o cualquier otro sistema o subsistema de decodificador descrito en esta memoria. En algunos ejemplos, el decodificador de vídeo 30 implementado por medio de circuitería lógica puede incluir un búfer de imágenes (por ejemplo, por medio de unidad(es) de procesamiento 420 o almacenamiento(s) en memoria 44)) y una unidad de procesamiento de gráficos (por ejemplo, por medio de unidad(es) de procesamiento 46). La unidad de procesamiento de gráficos puede acoplarse comunicativamente al búfer de imágenes. La unidad de procesamiento de gráficos puede incluir decodificador de vídeo 30 como implementado por medio de circuitería lógica 47 para materializar los diversos módulos como se discute con respecto a la FIG 3 y/o cualquier otro sistema o subsistema de codificador descrito en esta memoria.

En algunos ejemplos, la antena 42 del sistema de codificado de vídeo 40 se puede configurar para recibir un flujo de bits codificado de datos de vídeo. Como se discute, el flujo de bits codificado puede incluir datos, indicadores, valores de índice, datos de selección de modo, o algo semejante asociado con codificar un fotograma de vídeo como se discute en esta memoria, tal como datos asociados con la partición de codificado (por ejemplo, coeficientes de transformada o coeficientes de transformada cuantizados, indicadores opcionales (como se discute), y/o datos que definen la partición de codificado). El sistema de codificado de vídeo 40 puede incluir también puede incluir el decodificador de vídeo 30 acoplado a la antena 42 y configurado para decodificar el flujo de bits codificado. El dispositivo de exposición 45 se configura para presentar fotogramas de vídeo.

CODIFICADOR Y MÉTODO DE CODIFICACIÓN

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

Por ejemplo, la unidad de cálculo residual 204, la unidad de procesamiento de transformada 206, la unidad de cuantización 208, la unidad de procesamiento de predicción 260, y la unidad de codificación por entropía 270 forman

una ruta de señal directa del codificador 20, mientras por ejemplo la unidad de cuantización inversa 210, la unidad de procesamiento de transformada inversa 212, la unidad de reconstrucción 214, el búfer 216, el filtro de bucle 220, el búfer de imágenes decodificadas (DPB) 230, la unidad de procesamiento de predicción 260 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 se configura para recibir, por ejemplo por la entrada 202, una imagen 201 o un bloque 203 de la imagen 201, por ejemplo imagen de una secuencia de imágenes que forman un vídeo o una secuencia de vídeo. El bloque de imágenes 203 puede denominarse bloque de imágenes actual o bloque de imágenes a codificar, y la imagen 201 como imagen actual o imagen a codificar (en particular en codificado de vídeo para distinguir la imagen actual de otras imágenes, por ejemplo, imágenes previamente codificadas y/o decodificadas de la misma secuencia de vídeo, es decir, la secuencia de vídeo que también comprende la imagen actual).

PARTICIONADO

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

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

Como la imagen 201, el bloque 203 de nuevo es o se puede considerar como distribución o matriz bidimensional de muestras con valores de intensidad (valores de muestra), aunque de dimensión más pequeña que la imagen 201. En otras palabras, el bloque 203 puede comprender, por ejemplo, una distribución de muestras (por ejemplo una distribución luma en caso de una imagen monocromática 201) o tres distribuciones de muestras (por ejemplo distribuciones luma y dos croma en caso de una imagen a color 201) o cualquier otro número y/o clase de distribuciones dependiendo del formato de color aplicado. El número de muestras en dirección (o eje) horizontal y vertical del bloque 203 definen el tamaño del bloque 203.

El codificador 20 como se muestra en la Figura 2 se configura para codificar la imagen 201 bloque por bloque, por ejemplo la codificación y predicción se realiza por el bloque 203.

CÁLCULO DE RESIDUAL

La unidad de cálculo residual 204 se configura para calcular un bloque residual 205 en función del bloque de imagen 203 y un bloque de predicción 265 (además detalles acerca del bloque de predicción 265 se proporcionan más adelante), por ejemplo, al restar valores de muestra del bloque de predicción 265 de valores de muestra del bloque de imagen 203 muestra por muestra (píxel por píxel), para obtener el bloque residual 205 en el dominio de muestras.

TRANSFORMADA

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

La unidad de procesamiento de transformada 206 se puede configurar para aplicar aproximaciones enteras de DCT/DST, tales como transformadas especificadas para HEVC/H.265. Comparado con una transformada DCT ortogonal, tales aproximaciones a entero típicamente se escalan por cierto factor. A fin de preservar la norma del bloque residual que es procesado por transformadas hacia delante e inversa, se aplican factores de escalado adicionales como parte del proceso de transformada. Los factores de escalado se eligen típicamente en función de ciertas restricciones, como factores de escalado que son una potencia de dos para operación de desplazamiento, profundidad de bits de los coeficientes de transformada, compensación entre precisión y costes de implementación, etc. Factores de escalado específicos son, por ejemplo, especificados para la transformada inversa, por ejemplo por la unidad de procesamiento de transformada inversa 212, en un decodificador 30 (y la correspondiente transformada inversa, por ejemplo por la unidad de procesamiento de transformada inversa 212 en un codificador 20) y factores de escalado correspondientes para la transformada hacia delante, por ejemplo por la unidad de transformada de procesamiento 206, en un codificador 20 por consiguiente puede especificarse.

CUANTIZACIÓN

La unidad de cuantización 208 se configura para cuantizar los coeficientes de transformada 207 para obtener coeficientes de transformada cuantizados 209, por ejemplo, aplicando cuantización escalar o cuantización vectorial. Los coeficientes de transformada cuantizados 209 también pueden denominarse coeficientes residuales cuantizados 209. El proceso de cuantización puede reducir la profundidad de bits asociada con algunos o todos los coeficientes de transformada 207. Por ejemplo, un coeficiente de Transformada de n bits puede redondearse hacia abajo a un coeficiente de Transformada de m bits, donde n es mayor que m . El grado de cuantización puede modificarse ajustando un parámetro de cuantización (QP). Por ejemplo, para la cuantización escalar, se puede aplicar una escala diferente para lograr una cuantización más fina o más basta. Los tamaños de etapa de cuantización más pequeños corresponden a una cuantización más fina, mientras que los tamaños de etapa de cuantización más grandes corresponden a una cuantización más basta. El tamaño de etapa de cuantización aplicable puede indicarse mediante un parámetro de cuantización (QP). El parámetro de cuantización puede ser, por ejemplo, un índice de un conjunto predefinido de tamaños de etapa de cuantización aplicables. Por ejemplo, los parámetros de cuantización pequeños pueden corresponder a una cuantización fina (tamaños de etapas de cuantización pequeños) y los parámetros de cuantización grandes pueden corresponder a una cuantización basta (tamaños de etapas de cuantización grandes) o viceversa. La cuantización puede incluir división por un tamaño de etapa de cuantización y una descuantización correspondiente o inversa, por ejemplo mediante cuantización inversa 210, puede incluir multiplicación por el tamaño de etapa de cuantización. Las realizaciones según algunos estándares, por ejemplo HEVC, pueden configurarse para usar un parámetro de cuantización para determinar el tamaño de etapa de cuantización. Generalmente, el tamaño de etapa de cuantización se puede calcular sobre la base de un parámetro de cuantización usando una aproximación de punto fijo de una ecuación que incluye división. Se pueden introducir factores de escala adicionales para la cuantización y descuantización para restaurar la norma del bloque residual, que podría modificarse debido a la escala utilizada en la aproximación de punto fijo de la ecuación para el tamaño de etapa de cuantización y el parámetro de cuantización. En una implementación de ejemplo, se podría combinar el escalado de la transformada inversa y la descuantización. Alternativamente, pueden usarse tablas de cuantización personalizadas y señalarse desde un codificador a un decodificador, por ejemplo en un flujo de bits. La cuantización es una operación con pérdidas, en donde la pérdida aumenta al tamaños de etapa de cuantización crecientes.

La unidad de cuantización inversa 210 se configura para aplicar la cuantización inversa de la unidad de cuantización 208 sobre los coeficientes cuantizados para obtener coeficientes descuantizados 211, por ejemplo, aplicando la inversa del esquema de cuantización aplicado por la unidad de cuantización 208 en función o usando el mismo tamaño de etapa de cuantización que la unidad de cuantización 208. Los coeficientes descuantizados 211 también pueden denominarse coeficientes residuales descuantizados 211 y corresponden - aunque típicamente no son idénticos a los coeficientes de transformada debido a la pérdida por cuantización - a los coeficientes de transformada 207.

La unidad de procesamiento de transformada inversa 212 se configura para aplicar la transformada inversa de la transformada aplicada por la unidad de procesamiento de transformada 206, por ejemplo, una transformada de coseno discreta inversa (DCT) o transformada de seno discreta inversa (DST), para obtener un bloque de transformada inversa 213 en el dominio de muestra. El bloque de transformada inversa 213 también se puede denominar bloque descuantizado de transformada inversa 213 o bloque residual de transformada inversa 213.

La unidad de reconstrucción 214 (por ejemplo, Sumador 214) se configura para añadir el bloque de transformada inversa 213 (es decir, el bloque residual reconstruido 213) al bloque de predicción 265 para obtener un bloque reconstruido 215 en el dominio de muestras, por ejemplo añadiendo los valores de muestras del bloque residual reconstruido 213 y los valores de muestras del bloque de predicción 265.

Opcionalmente, una unidad de búfer 216 ("búfer" 216 por abreviar), por ejemplo, un búfer de línea 216, se configura para búfer o almacenar el bloque reconstruido 215 y los valores de muestras respectivos, por ejemplo, para intrapredicción. En realizaciones adicionales, el codificador se puede configurar para usar bloques reconstruidos no filtrados y/o los valores de muestra respectivos almacenados en unidad de búfer 216 para cualquier clase de estimación y/o predicción, por ejemplo intrapredicción.

Realizaciones del codificador 20 se pueden configurar de modo que, por ejemplo la unidad de búfer 216 no se usa únicamente para almacenar los bloques reconstruidos 215 para la unidad de intrapredicción 254 sino también para la unidad de filtro de bucle 220 (que mostrada en la Figura 2) y/o de modo que, por ejemplo, la unidad de búfer 216 y la unidad de búfer de imágenes decodificadas 230 forman un búfer. Realizaciones adicionales se pueden configurar para usar bloques filtrados 221 y/o bloques o muestras del búfer de imágenes decodificadas 230 (no se muestran en la Figura 2) como entrada o base para intrapredicción 254.

La unidad de filtro de bucle 220 (o abreviado "filtro de bucle" 220) se configura para filtrar el bloque reconstruido 215 para obtener un bloque filtrado 221, por ejemplo para suavizar transiciones de píxeles o mejorar de otro modo la calidad de video. La unidad de filtro de bucle 220 está pensada para representar uno o más filtros de bucle tales como un filtro de des-bloqueo, un filtro de desplazamiento adaptativo por muestra (SAO), u otros filtros, por ejemplo, un filtro bilateral, un filtro de bucle adaptativo (ALF), un filtro de agudizamiento o suavización, o filtros colaborativos. Aunque la unidad de filtro de bucle 220 se muestra en la FIGURA 2 como que es un filtro de bucle, en otras configuraciones, la unidad de filtro de bucle 220 puede implementarse como filtro posbucle. El bloque filtrado 221 también se puede denominar bloque filtrado reconstruido 221. El búfer de imágenes decodificadas 230 puede almacenar los bloques de

codificado reconstruidos después de que la unidad de filtro de bucle 220 realice la operación de filtrado en los bloques de codificado reconstruidos.

5 Realizaciones del codificador 20 (respectivamente la unidad de filtro de bucle 220) se pueden configurar para tener como salida parámetros de filtro de bucle (tales como información de desplazamiento adaptativo de muestras), por ejemplo, directamente o codificado por entropía a través de la unidad de codificado por entropía 270 o cualquier otra unidad de codificado por entropía, de modo que, por ejemplo, un decodificador 30 puede recibir y aplicar los mismos parámetros de filtro de bucle para decodificar.

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

25 La unidad de procesamiento de predicción 260, también denominada unidad de procesamiento de predicción de bloques 260, se configura para recibir u obtener el bloque 203 (bloque actual 203 de la imagen actual 201) y datos de imagen reconstruida, por ejemplo, muestras de referencia de la misma imagen (actual) del búfer 216 y/o datos de imagen de referencia 231 de una o una pluralidad de imágenes decodificadas previamente del búfer de imágenes decodificadas 230, y procesar tales datos para predicción, es decir, para proporcionar un bloque de predicción 265 que puede ser un bloque de interpredicción 245 o un bloque de intrapredicción 255.

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

35 Realizaciones de la unidad de selección de modo 262 se pueden configurar para seleccionar el modo de predicción (por ejemplo, de modos de predicción soportados por la unidad de procesamiento de predicción 260), que proporciona la mejor coincidencia o en otras palabras el residual mínimo (el residual mínimo significa mejor compresión para transmisión o almacenamiento), o una sobrecarga de señalización mínima (sobrecarga de señalización mínima significan mejor compresión para transmisión o almacenamiento), o que considera o equilibra ambos. La unidad de selección de modo 262 se puede configurar para determinar el modo de predicción en función de optimización de distorsión de tasa (RDO), es decir, seleccionar el modo de predicción que proporciona una optimización mínima distorsión de tasa o cuya distorsión de tasa asociada satisface al menos un criterio de selección de modo de predicción.

45 A continuación se explicará más en detalle el procesamiento por predicción (por ejemplo unidad de procesamiento de predicción 260 y selección de modo (por ejemplo por la unidad de selección de modo 262) realizada por un ejemplo de codificador 20.

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

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

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

65 Adicional a los modos de predicción anteriores, puede aplicarse modo omitir y/o modo directo.

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

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

El codificador 20 se puede, por ejemplo, configurar para seleccionar un bloque de referencia de una pluralidad de bloques de referencia de una misma o diferentes imágenes de la pluralidad de otras imágenes, y proporcionar una imagen de referencia (o índice de imágenes de referencia...) y/o un desplazamiento (desplazamiento espacial) entre una 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 2). Este desplazamiento también se llama vector de movimiento (MV).

La unidad de compensación de movimiento se configura para obtener, por ejemplo recibir, un parámetro de interpredicción, y realizar interpredicción en función o usando el parámetro de interpredicción para obtener un bloque de interpredicción 245. La compensación de movimiento, realizada por la unidad de compensación de movimiento (no mostrada en la figura 2) puede implicar reunir o generar el bloque de predicción en función de un vector de movimiento/bloque determinado a través de estimación de movimiento, posiblemente realizando interpolación en precisión subpíxel. El filtrado por interpolación puede generar muestras de píxeles adicionales de muestras de píxeles conocidas, aumentando así potencialmente el número de bloques de predicción candidatos que se pueden usar para codificar un bloque de imagen. Al recibir el vector de movimiento para la PU del bloque de imagen actual, la unidad de compensación de movimiento 246 puede localizar el bloque de predicción al que el vector de movimiento apunta en una de las listas de imágenes de referencia. La unidad de compensación de movimiento 246 también puede generar elementos de sintaxis asociados con los bloques y el sector de vídeo para su utilización por el decodificador de vídeo 30 para decodificar los bloques de imagen del sector de vídeo.

La unidad de intrapredicción 254 se configura para obtener, por ejemplo recibir, el bloque de imagen 203 (el bloque de imagen actual) y uno o una pluralidad de bloques reconstruidos previamente, por ejemplo bloques vecinos reconstruidos, de una misma imagen para intraestimación. El codificador 20 se puede, por ejemplo, configurar para seleccionar un modo de intrapredicción de una pluralidad de modos de intrapredicción (predeterminados).

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

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

La unidad de codificado por entropía 270 se configura para aplicar un algoritmo o esquema de codificado por entropía (por ejemplo, un esquema de codificado de longitud variable (VLC), un esquema de VLC adaptativo al contexto (CAVLC), un esquema de codificado aritmético, una codificado aritmético binario adaptativo al contexto (CABAC), codificado aritmético binario adaptativo al contexto basado en sintaxis (SBAC), codificado por entropía de partición de intervalo de probabilidad (PIPE) u otra metodología o técnica de codificado por entropía) en los coeficientes residuales cuantizados 209, parámetros de interpredicción, parámetros de intrapredicción y/o parámetros de filtro de bucle individualmente o conjuntamente (o nada en absoluto) para obtener datos de imagen codificados 21 que pueden salir a través de la salida 272, por ejemplo, en forma de un flujo de bits codificado 21. El flujo de bits codificado 21 puede transmitirse al decodificador de vídeo 30, o archivarse para transmisión o recuperación posterior por el decodificador de vídeo 30. La unidad de codificación por entropía 270 puede configurarse además para codificar por entropía otros elementos de sintaxis para un sector de vídeo actual que se está codificando.

Otras variaciones estructurales del codificador de vídeo 20 se pueden usar para codificar el flujo de vídeo. Por ejemplo, un codificador basado en no transformada 20 puede cuantizar la señal residual directamente sin la unidad de

procesamiento de transformada 206 para ciertos bloques o fotogramas. En otra implementación, un codificador 20 puede tener la unidad de cuantización 208 y la unidad de cuantización inversa 210 combinadas en una sola unidad.

La Figura 3 muestra un decodificador de vídeo 30 ejemplar que se configura para implementar las técnicas de esta presente solicitud. El decodificador de vídeo 30 configurado para recibir datos de imagen codificados (por ejemplo codificado flujo de bits) 21, por ejemplo codificado por el codificador 100, para obtener una imagen decodificada 131. Durante el proceso de descodificación, el decodificador de vídeo 30 recibe datos de vídeo, por ejemplo un flujo de bits de vídeo codificado que representa bloques de imagen de un codificado sector de vídeo y elementos de sintaxis asociado, del codificador de vídeo 100.

En el ejemplo de la figura 3, el decodificador 30 comprende una unidad de unidad de decodificación por entropía 304, una unidad de cuantización inversa 310, una unidad de procesamiento de transformada inversa 312, una unidad de reconstrucción 314 (por ejemplo una sumador 314), un búfer 316, un filtro de bucle 320, un búfer de imágenes decodificadas 330, y una unidad de procesamiento de predicción 360. La unidad de procesamiento de predicción 360 puede incluir una unidad de interpredicción 344, una unidad de intrapredicción 354 y una unidad de selección de modo 362. El decodificador de vídeo 30 puede, en algunos ejemplos, realizar una etapa de decodificación generalmente recíproca a la etapa de codificación descrita con respecto al codificador de vídeo 100 de la FIGURA 2.

La unidad de decodificación por entropía 304 se configura para realizar decodificación por entropía a los datos de imagen codificados 21 para obtener, por ejemplo, coeficientes cuantizados 309 y/o parámetros de codificado decodificados (no mostrados en la Figura 3), por ejemplo (decodificados) cualquiera o todos los parámetros de interpredicción, parámetros de intrapredicción, parámetros de filtro de bucle y/u otros elementos de sintaxis. La unidad de decodificación por entropía 304 se configura además para reenviar parámetros de interpredicción, parámetros de intrapredicción y/u otros elementos de sintaxis a la unidad de procesamiento de predicción 360. El decodificador de vídeo 30 puede recibir los elementos de sintaxis a nivel de sector de vídeo y/o nivel de bloque de imagen.

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

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

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

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

La unidad de cuantización inversa 310 se configura para cuantizar a la inversa, es decir, decuantizar, los coeficientes de transformada cuantizados proporcionados en el flujo de bits y decodificados por la unidad de decodificación por entropía 304. El proceso de cuantización inversa puede incluir el uso de un parámetro de cuantización calculado por

el codificador de vídeo 100 para cada bloque de vídeo en el sector de vídeo para determinar un grado de cuantización y, de la misma manera, un grado de cuantización inversa que debería aplicarse.

La unidad de procesamiento de transformada inversa 312 se configura para aplicar una transformada inversa, por ejemplo, una DCT inversa, una transformada de enteros inversa, o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada, para producir bloques residuales en el dominio de píxeles.

La unidad de reconstrucción 314 (por ejemplo, Sumador 314) se configura para añadir el bloque de transformada inversa 313 (es decir, el bloque residual reconstruido 313) al bloque de predicción 365 para obtener un bloque reconstruido 315 en el dominio de muestras, por ejemplo añadiendo los valores de muestras del bloque residual reconstruido 313 y los valores de muestras del bloque de predicción 365.

La unidad de filtro de bucle 320 (ya sea en el bucle de codificado o tras el bucle de codificado) se configura para filtrar el bloque reconstruido 315 para obtener un bloque filtrado 321, por ejemplo para suavizar transiciones de píxeles o mejorar de otro modo la calidad de vídeo. En un ejemplo, la unidad de bucle filtro 320 se puede configurar para realizar cualquier combinación de las técnicas de filtrado descritas más adelante. La unidad de filtro de bucle 320 está pensada para representar uno o más filtros de bucle tales como un filtro de des-bloqueo, un filtro de desplazamiento adaptativo por muestra (SAO), u otros filtros, por ejemplo, un filtro bilateral, un filtro de bucle adaptativo (ALF), un filtro de agudización o suavización, o filtros colaborativos. Aunque la unidad de filtro de bucle 320 se muestra en la FIGURA 3 como que es un filtro de bucle, en otras configuraciones, la unidad de filtro de bucle 320 puede implementarse como filtro posbucle.

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

El decodificador 30 se configura para tener como salida la imagen decodificada 331, por ejemplo, por medio de una salida 332, para presentación o visualización a un usuario.

Otras variaciones del decodificador de vídeo 30 se pueden usar para decodificar el flujo de bits comprimido. Por ejemplo, el decodificador 30 puede producir el flujo de vídeo de salida sin la unidad de filtrado de bucle 320. Por ejemplo, un decodificador basado en no transformada 30 puede cuantizar a la inversa la señal residual directamente sin la unidad de procesamiento de transformada inversa 312 para ciertos bloques o fotogramas. En otra implementación, el decodificador de vídeo 30 puede tener la unidad de cuantización inversa 310 y la unidad de procesamiento de transformada inversa 312 combinadas en una única unidad.

La FIG 4 es un diagrama esquemático de un dispositivo de codificado de vídeo 400 según una realización de la divulgación. El dispositivo de codificado de vídeo 400 es adecuado para implementar las realizaciones divulgadas que se describen en esta memoria. En una realización, el dispositivo de codificado de vídeo 400 puede ser un decodificador tal como el decodificador de vídeo 30 de la FIGURA 1A o un codificador tal como el codificador de vídeo 20 de la FIGURA 1A. En una realización, el dispositivo de codificado de vídeo 400 puede ser uno o más componentes del decodificador de vídeo 30 de la FIGURA 1A o el codificador de vídeo 20 de la FIGURA 1A como se ha descrito anteriormente.

El dispositivo de codificado de vídeo 400 comprende puertos de entrada 410 y unidades de receptor (Rx) 420 para recibir datos; un procesador, unidad lógica, o Unidad de procesamiento central (CPU) 430 para procesar los datos; unidades de transmisor (Tx) 440 y puertos de salida 450 para transmitir los datos; y una memoria 460 para almacenar los datos. El dispositivo de codificado de vídeo 400 también puede comprender componentes óptico-a-eléctrico y componentes eléctrico-a-óptico (EO) que se acoplan a los puertos de entrada 410, las unidades de receptor 420, las unidades de transmisor 440 y los puertos de salida 450, para salida o entrada de señales ópticas o eléctricas.

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

La memoria 460 comprende uno o más discos, unidades de cinta y unidades de estado sólido y se puede usar como dispositivo de almacenamiento de datos de desbordamiento, para almacenar programas cuando tales programas se seleccionan para ejecución, y para almacenar instrucciones y datos que se leen durante ejecución de programa. La memoria 460 puede ser volátil y/o no volátil, y puede ser una memoria de solo lectura (ROM), una memoria de acceso

aleatorio (RAM), una memoria accesible por dirección de contenido ternario (TCAM) y/o una memoria de acceso aleatorio estática (SRAM).

La Figura 5 es un diagrama de bloques simplificado de un aparato 500 que se puede usar como uno cualquiera o ambos del dispositivo fuente 310 y el dispositivo de destino 320 en la Figura 1A según una realización ejemplar. El aparato 500 puede implementar técnicas de esta presente solicitud descritas anteriormente. El aparato 500 puede ser en forma de sistema informático que incluye múltiples dispositivos de cómputo, o en forma de un único dispositivo informático, por ejemplo, un teléfono móvil, una tableta, un ordenador portátil, un ordenador notebook, un ordenador de escritorio, y similares.

Un procesador 502 en el aparato 500 puede ser una unidad de procesamiento central. Como alternativa, el procesador 502 puede ser cualquier otro tipo de dispositivo, o múltiples dispositivos, capaces de manipular o procesar información existente ahora o desarrollada más adelante. Aunque las implementaciones descritas se pueden poner en práctica con un único procesador como se muestra, por ejemplo, el procesador 502, se pueden lograr ventajas en velocidad y eficiencia usando más de un procesador.

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 puede usarse 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 de aplicación 510, los programas de aplicación 510 incluyen al menos un programa que permite al procesador 502 realizar los métodos descritos aquí. Por ejemplo, los programas de aplicación 510 pueden incluir aplicaciones 1 a N, que incluyen además una aplicación de codificado de vídeo que realiza los métodos descritos aquí. El aparato 500 también puede incluir memoria adicional en forma de almacenamiento secundario 514, que puede, por ejemplo, ser una tarjeta de memoria usada con un dispositivo informático móvil. Como las sesiones de comunicación de vídeo pueden contener una cantidad significativa de información, pueden almacenarse en total o en parte en el almacenamiento secundario 514 y cargarse en la memoria 504 según sea necesario para procesar.

El aparato 500 también puede incluir uno o más dispositivos de salida, tal como una pantalla 518. La pantalla 518 puede ser, en un ejemplo, una pantalla sensible al tacto que combina una pantalla y un elemento sensible al tacto que puede funcionar para detectar una entradas táctiles. La pantalla 518 se puede acoplar al procesador 502 por medio del bus 512. Otros dispositivos de salida que permiten a un usuario programar o usar de otro modo el aparato 500 se pueden proporcionar además o como alternativa a la pantalla 518. Cuando el dispositivo de salida es o incluye una pantalla, la pantalla se puede implementar de diversas maneras, que incluye por una pantalla de cristal líquido (LCD), una pantalla de tubo de rayos catódicos (CRT), una pantalla de plasma o una pantalla de diodos emisores de luz (led), tal como una pantalla de ledes orgánicos (OLED).

El aparato 500 también puede incluir o estar en comunicación con un dispositivo detector de imágenes 520, por ejemplo una cámara, o cualquier otro dispositivo detector de imágenes 520 existente ahora o desarrollado en adelante que pueda detectar una imagen tal como la imagen de un usuario que maneja el aparato 500. El dispositivo detector de imágenes 520 puede posicionarse de manera que se dirige hacia el usuario que maneja el aparato 500. En un ejemplo, la posición y el eje óptico del dispositivo detector de imágenes 520 se pueden configurar de manera que el campo de visión incluye un área que está directamente adyacente a la pantalla 518 y desde la que es visible la pantalla 518.

El aparato 500 también puede incluir o estar en comunicación con un dispositivo detector de sonido 522, por ejemplo un micrófono, o cualquier otro dispositivo detector de sonido existente ahora o desarrollado en adelante que pueda detectar sonidos cerca del aparato 500. El dispositivo detector de sonido 522 se puede posicionar de manera que se dirija hacia el usuario que maneja el aparato 500 y se puede configurar para recibir sonidos, por ejemplo, discurso u otras declaraciones, hecho por el usuario mientras el usuario maneja el aparato 500.

Aunque la FIG 5 representa el procesador 502 y la memoria 504 del aparato 500 como integrados en una única unidad, se pueden utilizar otras configuraciones. Las operaciones del procesador 502 se pueden distribuir por múltiples máquinas (cada máquina que tiene uno o más de procesadores) que se pueden acoplar directamente o por un área local u otra red. La memoria 504 se puede distribuir por múltiples máquinas tales como una memoria basada en red o memoria en múltiples máquinas que realizan las operaciones del aparato 500. Aunque aquí se representa como único bus, el bus 512 del aparato 500 se puede componer de múltiples buses. Además, el almacenamiento secundario 514 se puede acoplar directamente a los otros componentes del aparato 500 o se puede acceder por medio de una red y puede comprender una única unidad integrada tal como una tarjeta de memoria o múltiples unidades tales como múltiples tarjetas de memoria. El aparato 500 puede así implementarse en una gran variedad de configuraciones.

El codificado de vídeo como H.265/HEVC o H.266/VVC se basa en una estructura híbrida de predicción más transformada. La unidad de árbol de codificado (CTU) se divide en a una o más unidades de codificado (CU). Una CU puede contener un bloque luma y dos bloques croma (por ejemplo, formato YUV4:2:0 o YUV4:4:4), o una CU puede únicamente contener un bloque luma, o una CU puede únicamente contener dos bloques croma. Una CU puede corresponder a una unidad de predicción (PU) y una unidad de transformada (TU), o puede corresponder a una unidad

de predicción y múltiples unidades de transformada. La unidad de predicción contiene bloques de predicción de muestras de predicción, y la unidad de transformada contiene bloques de transformada de muestras residuales. Las muestras reconstruidas de la CU se obtienen añadiendo las muestras de predicción con correspondientes muestras residuales.

En el codificador, la información de predicción (por ejemplo, modo de predicción y vector de movimiento información) e información de residual (por ejemplo, modo de transformada, coeficientes de transformada y parámetro de cuantización) de una CU se deciden por optimización de distorsión de tasa y luego codificando en flujo de bits. En el decodificador, se analiza sintácticamente la información de predicción y la información de residual. Según la información de predicción, se realiza intra- o interpredicción para generar las muestras de predicción de la unidad de predicción. Según la información de residual, se realiza cuantización inversa y transformada inversa para generar muestras residuales de la unidad de transformada. Las muestras de predicción y las muestras residuales se añaden para obtener las muestras reconstruidas de la unidad de codificado.

Un modo de transformada básico es el residual de una CU que se transforma en una TU, es decir, tamaño de CU igual a tamaño de TU. En Árbol Cuádruple de Residual (RQT) en HEVC, una CU puede dividirse en 4 TU de igual tamaño, y la información de si cada TU tiene residual o no se señala por marcadores de bloques codificados (cbf) de cada TU. La información de si la CU tiene residual se señala por raíz cbf de la CU (por ejemplo, el elemento de sintaxis `rqt_root_cbf` en HEVC).

Transformada Adaptativa de núcleo Múltiple (AMT) en VTM permite un bloque de transformada que usa un tipo de transformada entre un conjunto de tipos de transformada. El tipo de transformada se señala en flujo de bits, y un tipo de transformada especifica la transformada horizontal (o transformada de fila) y transformada vertical (o transformada de columna) del bloque de transformada. Las posibles transformadas núcleo para la transformada horizontal/vertical incluyen Transformada de Seno Discreta (DST) y Transformada Discreta de Coseno (DCT), por ejemplo DST-7 (tipo de DST 7), DST-1, DCT-8, DCT-2 y DCT5.

Para mejorar la eficiencia de codificado de transformadas, en la presente invención se introducen nuevos modos de transformada para una CU. Para cada uno de los nuevos modos de transformada, una CU se pone en mosaico en N ($N=2, 3$ o 4) TU, y únicamente una TU de la CU tiene residual y las N-1 TU restantes de la CU se infieren para no tener residual. Además, el tipo de transformada (es decir, la transformada horizontal y la vertical) de la TU con residual se obtiene o selecciona de un conjunto de tipos de transformada especificado por el modo de transformada. En otras palabras, cada uno de los nuevos modos de transformada especifica TU particionado, la única TU con residual, y el conjunto de tipos de transformada de la TU con residual.

Una realización de la presente divulgación describe el proceso de decodificar una unidad de codificado. Se decodifica un flujo de bits que contiene al menos una imagen de datos de vídeo. Una imagen se parte en una pluralidad de regiones de imagen rectangulares y cada región corresponde a una unidad de Árbol de Codificado (CTU). Una CTU se particiona en una pluralidad de unidades de codificado, según información de partición de bloque contenido en el flujo de bits. La información de codificado de las unidades de codificado se analiza sintácticamente del flujo de bits, y píxeles de las unidades de codificado se reconstruyen en función de la información de codificado. La información de codificado contiene información de predicción e información de transformada.

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 además contener modo de intrapredicción, tal como el modo plano, modo DC y modo angular en HEVC y VVC. Si el modo de predicción es interpredicción, la información de predicción puede además contener si usar modo omitir o modo de fusión o modo AMVP (predicción avanzada de vector de movimiento), índice de candidato de fusión, dirección de interpredicción, índice de fotograma de referencia, vector de movimiento, si usar predicción basada en modelo afín, etc.

La información de transformada indica el modo de transformada de la CU y los coeficientes de transformada asociados con una o más TU en la CU. Aparte del modo de transformada convencional 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 transformada. Para cada uno de los nuevos modos de transformada, una CU se pone en mosaico en N ($N=2, 3$ o 4) TU, y únicamente una TU tiene residual y la N-1 TU restantes se infieren para no tener residual. Además, el tipo de transformada (es decir, la transformada horizontal y la vertical) de la TU con residual es especificado por el modo de transformada.

En un ejemplo, los nuevos modos de transformada 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 $W \times H$ se divide en 4 TU de tamaño $(W/2) \times (H/2)$, es decir la TU0 superior-izquierda, la TU1 superior-derecha, la TU2 inferior-izquierda y la TU3 inferior-derecha. Únicamente una de la TU0, TU1, TU2 y TU3 tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada (es decir, el conjunto de tipos de transformada son correspondientes al modo de transformada). El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador o un índice.

1) Modo de transformada Q0: TU0 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-8 y DCT-8, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-7 y DST-7, respectivamente.

2) Modo de transformada Q1: TU1 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-7 y DCT-8, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-8 y DST-7, respectivamente.

3) Modo de transformada Q2: TU2 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-8 y DST-7, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-7 y DCT-8, respectivamente.

4) Modo de transformada Q3: TU3 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU3 son DST-7 y DST-7, respectivamente, un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU3 son DCT-8 y DCT-8, respectivamente.

En este ejemplo, el segundo tipo de transformada es dependiente del primer tipo de transformada. Si la transformada horizontal del primer tipo de transformada es DCT-8 (o DST-7), la transformada horizontal del segundo tipo de transformada es DST-7 (o DCT-8). Si la transformada vertical del primer tipo de transformada es DCT-8 (o DST-7), la transformada vertical del segundo tipo de transformada es DST-7 (o DCT-8).

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para los modos de transformada Q0, Q1, Q2, y Q3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre Q0, Q1, Q2 y Q3. El tipo de transformada usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por dos marcadores o señalado por código unario truncado. Los cuatro tipos de transformada son de la siguiente manera:

Tipo de transformada 1: una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DST-7, denotada como DST-7/DST-7;

Tipo de transformada 2: una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-8, denotada como DST-7/DCT-8;

Tipo de transformada 3: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7, denotada como DCT-8/DST-7;

Tipo de transformada 4: una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DCT-8, denotada como DCT-8/DCT-8.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en un ejemplo, los nuevos modos de transformada 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 $W \times H$ se divide en 4 TU de tamaño $(W/2) \times (H/2)$, es decir la TU0 superior-izquierda, la TU1 superior-derecha, la TU2 inferior-izquierda y la TU3 inferior-derecha. Únicamente una de la TU0, TU1, TU2 y TU3 tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada (es decir, el conjunto de tipos de transformada son correspondientes al modo de transformada). El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador o un índice.

1) Modo de transformada Q0: TU0 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-4 y DCT-4, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-4 y DST-4, respectivamente.

2) Modo de transformada Q1: TU1 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-4 y DCT-4, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-4 y DST-4, respectivamente.

3) Modo de transformada Q2: TU2 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-4 y DCT-4, respectivamente, y un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-4 y DST-4, respectivamente.

4) Modo de transformada Q3: TU3 tiene residual; un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU3 son DST-4 y DST-4, respectivamente, un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU3 son DCT-4 y DCT-4, respectivamente.

En este ejemplo, el segundo tipo de transformada es dependiente del primer tipo de transformada. Si la transformada horizontal del primer tipo de transformada es DCT-4 (o DST-4), la transformada horizontal del segundo tipo de transformada es DST-4 (o DCT-4). Si la transformada vertical del primer tipo de transformada es DCT-4 (o DST-4), la transformada vertical del segundo tipo de transformada es DST-4 (o DCT-4).

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para los modos de transformada Q0, Q1, Q2, y Q3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre Q0, Q1, Q2 y Q3. El tipo de transformada usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por dos marcadores o señalado por código unario truncado. Los cuatro tipos de transformada son de la siguiente manera:

- Tipo de transformada 1: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DST-4, denotada como DST-4/DST-4;
- Tipo de transformada 2: una transformada horizontal de la TU residual es DST-4 y una transformada vertical de la TU residual es DCT-4, denotada como DST-4/DCT-4;
- Tipo de transformada 3: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DST-4, denotada como DCT-4/DST-4;
- Tipo de transformada 4: una transformada horizontal de la TU residual es DCT-4 y una transformada vertical de la TU residual es DCT-4, denotada como DCT-4/DCT-4.

Para la TU que tiene residual, marcadores de bloques codificados (cbf) de tres componentes, es decir, los componentes Cb, Cr e Y, pueden analizarse sintácticamente 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, los cbfs de todos los componentes de color se infieren para ser 0, es decir, no se analizan sintácticamente del flujo de bits.

En otra realización, los nuevos modos de transformada 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 únicamente la TU de dimensiones más pequeñas tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HQ0: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/4)$ y TU1 de tamaño $W \times (3H/4)$; la TU0 superior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2, y la transformada vertical de TU0 es DCT-8; un segundo tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2, y la transformada vertical de TU0 es DST-7. Como ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es siempre DST-7or es siempre DCT-2. Como otro ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del primer tipo de transformada de TU0 es DST-7 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es siempre DCT-8 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del segundo tipo de transformada de TU0 es DCT-8 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize.

2) Modo de transformada HQ1: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (3H/4)$ y TU1 de tamaño $W \times (H/4)$; la TU1 inferior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2, y la transformada vertical de TU1 es DST-7; un segundo tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2, y la transformada vertical de TU1 es DCT-8;

3) Modo de transformada VQ0: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(W/4) \times H$ y TU1 de tamaño $(3W/4) \times H$; la TU0 izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es DCT-8, y la transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU0 es DST-7, y la transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2. Como ejemplo, la transformada vertical del primer tipo de transformada de TU0 es siempre DST-7 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del

primer tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del primer tipo de transformada de TU0 es DST-7 si la altura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es siempre DCT-8 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del segundo tipo de transformada de TU0 es DCT-8 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

4) Modo de transformada VQ1: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(3 W/4) \times H$ y TU1 de tamaño $(W/4) \times H$; la TU1 derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es DST-7, y la transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU1 es DCT-8, y la transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para HQ0, HQ1, VQ0, y VQ1.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-7/DST-7, DST-7/DCT-8, DCT-8/DST-7, y DCT-8/DCT-8, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HQ0, HQ1, VQ0, y VQ1.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, los nuevos modos de transformada 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 únicamente la TU de dimensiones más pequeñas tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HQ0: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/4)$ y TU1 de tamaño $W \times (3H/4)$; la TU0 superior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2, y la transformada vertical de TU0 es DCT-4; un segundo tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2, y la transformada vertical de TU0 es DST-4. Como ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es siempre DST-4 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del primer tipo de transformada de TU0 es DST-4 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es siempre DCT-4 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del segundo tipo de transformada de TU0 es DCT-4 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize.

2) Modo de transformada HQ1: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (3H/4)$ y TU1 de tamaño $W \times (H/4)$; la TU1 inferior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2, y la transformada vertical de TU1 es DST-4; un segundo tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2, y la transformada vertical de TU1 es DCT-4.

3) Modo de transformada VQ0: la CU se divide por un vertical partir en dos TU, TU0 de tamaño $(W/4) \times H$ y TU1 de tamaño $(3 W/4) \times H$; la TU0 izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es DCT-4, y la transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU0 es DST-4, y la transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2. Como ejemplo, la transformada vertical del primer tipo de transformada de TU0 es siempre DST-4 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del primer tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del primer tipo de transformada de TU0 es DST-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es siempre DCT-4 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral

MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del segundo tipo de transformada de TU0 es DCT-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

4) Modo de transformada VQ1: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(3W/4) \times H$ y TU1 de tamaño $(W/4) \times H$; la TU1 derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es DST-4, y la transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU1 es DCT-4, y la transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para HQ0, HQ1, VQ0, y VQ1.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-4/DST-4, DST-4/DCT-4, DCT-4/DST-4, y DCT-4/DCT-4, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HQ0, HQ1, VQ0, y VQ1.

Se debe observar que una transformada DST o DCT se puede implementar usando multiplicación de matriz, o se puede implementar usando estructura de mariposa parcial, o se puede implementar como filtrado por ajuste de transformada con transformada DCT-2 inversa como se introduce en JVET-L0386. El método en JVET-L0386 es una manera para aproximar un objetivo de tipo de DST (por ejemplo, DST-7) o un objetivo de tipo de DCT (por ejemplo, DCT-8 o DCT-4) con un filtro de ajuste como preprocesamiento e inversa de DCT-2.

En una realización, los nuevos modos de transformada 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 únicamente una TU tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada (es decir el conjunto de tipos de transformada) son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HH0: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU0 superior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2, y la transformada vertical de TU0 es DCT-8; un segundo tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2, y la transformada vertical de TU0 es DST-7. Como ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es siempre DST-7 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del primer tipo de transformada de TU0 es DST-7 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es siempre DCT-8 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del segundo tipo de transformada de TU0 es DCT-8 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize.

2) Modo de transformada HH1: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU1 inferior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2, y transformada vertical de TU1 es DST-7; un segundo tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2, y transformada vertical de TU1 es DCT-8.

3) Modo de transformada VH0: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU0 izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es DCT-8, y transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU0 es DST-7, y transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2. Como ejemplo, la transformada vertical del primer tipo de transformada de TU0 es siempre DST-7 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del primer tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del primer tipo de transformada de TU0 es DST-7 si la altura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es siempre DCT-8 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral

MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del segundo tipo de transformada de TU0 es DCT-8 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

4) Modo de transformada VH1: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU1 derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es DST-7, y transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-7 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU1 es DCT-8, y transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-8 o DCT-2.

En un ejemplo de implementación alternativa que no comprende todos los rasgos necesarios para implementar la presente invención, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en un ejemplo, los nuevos modos de transformada 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 únicamente una TU tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada (es decir el conjunto de tipos de transformada) son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HH0: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU superior0 tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2, y la transformada vertical de TU0 es DCT-4; un segundo tipo de transformada es que la transformada horizontal de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2, y la transformada vertical de TU0 es DST-4. Como ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es siempre DST-4 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del primer tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del primer tipo de transformada de TU0 es DST-4 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es siempre DCT-4 o es siempre DCT-2. Como otro ejemplo, la transformada horizontal del segundo tipo de transformada de TU0 es DCT-2 si la anchura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada horizontal del segundo tipo de transformada de TU0 es DCT-4 si la anchura de TU0 es igual o menor que el umbral MaxMtsSize.

2) Modo de transformada HH1: la CU se divide por una partición horizontal en dos TU, TU0 de tamaño $W \times (H/2)$ y TU1 de tamaño $W \times (H/2)$; la TU1 inferior tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2, y transformada vertical de TU1 es DST-4; un segundo tipo de transformada es que la transformada horizontal de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2, y transformada vertical de TU1 es DCT-4.

3) Modo de transformada VH0: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU0 izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU0 es DCT-4, y transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU0 es DST-4, y transformada vertical de TU0 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2. Como ejemplo, la transformada vertical del primer tipo de transformada de TU0 es siempre DST-4 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del primer tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del primer tipo de transformada de TU0 es DST-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize. Como ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es siempre DCT-4 o es siempre DCT-2. Como otro ejemplo, la transformada vertical del segundo tipo de transformada de TU0 es DCT-2 si la altura de TU0 es mayor que un umbral MaxMtsSize (por ejemplo, MaxMtsSize = 32), y la transformada vertical del segundo tipo de transformada de TU0 es DCT-4 si la altura de TU0 es igual o menor que el umbral MaxMtsSize.

4) Modo de transformada VH1: la CU se divide por una partición vertical en dos TU, TU0 de tamaño $(W/2) \times H$ y TU1 de tamaño $(W/2) \times H$; la TU1 derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal de TU1 es DST-4, y transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DST-4 o DCT-2; un segundo tipo de transformada es que la transformada horizontal de TU1 es DCT-4, y transformada vertical de TU1 es una transformada de núcleo predefinida, por ejemplo, DCT-4 o DCT-2.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para HH0, HH1, VH0, y VH1.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-7/DST-7, DST-7/DCT-8, DCT-8/DST-7, y DCT-8/DCT-8, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HH0, HH1, VH0, y VH1.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-4/DST-4, DST-4/DCT-4, DCT-4/DST-4, y DCT-4/DCT-4, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HH0, HH1, VH0, y VH1.

En otro ejemplo, los nuevos modos de transformada 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 únicamente la TU de medio tamaño de la CU tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HH2: la CU se divide en horizontal dirección 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; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-7 y DCT-2 (o DST-7 y DST-1), respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-8 y DCT-2 (o DCT-8 y DST-1), respectivamente.

2) Modo de transformada VH2: la CU se divide en vertical dirección 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; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-2 y DST-7 (o DST-1 y DST-7), respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-2 y DCT-8 (o DST-1 y DCT-8), respectivamente.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para HH2 y VH2.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-7/DST-7, DST-7/DCT-8, DCT-8/DST-7, y DCT-8/DCT-8, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HH2 y VH2.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, los nuevos modos de transformada 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 únicamente la TU de medio tamaño de la CU tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada HH2: la CU se divide en horizontal dirección 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; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-4 y DCT-2 (o DST-4 y DST-1), respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-4 y DCT-2 (o DCT-4 y DST-1), respectivamente.

2) Modo de transformada VH2: la CU se divide en vertical dirección 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; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-2 y DST-4 (o DST-1 y DST-4), respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-2 y DCT-4 (o DST-1 y DCT-4), respectivamente.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para HH2 y VH2.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-4/DST-4, DST-4/DCT-4, DCT-4/DST-4, y DCT-4/DCT-4, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre HH2 y VH2.

En otro ejemplo, los nuevos modos de transformada 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. Únicamente una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada QH0: la TU0 superior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-8 y DCT-8, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-7 y DST-7, respectivamente.

2) Modo de transformada QH1: la TU1 superior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-7 y DCT-8, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-8 y DST-7, respectivamente.

3) Modo de transformada QH2: la TU1 inferior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-8 y DST-7, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-7 y DCT-8, respectivamente.

4) Modo de transformada QH3: la TU2 inferior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-7 y DST-7, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-8 y DCT-8, respectivamente.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para QH0, QH1, QH2, y QH3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-7/DST-7, DST-7/DCT-8, DCT-8/DST-7, y DCT-8/DCT-8, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre QH0, QH1, QH2, y QH3.

En un ejemplo de implementación alternativa que no comprende todos los rasgos necesarios para implementar la presente invención, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, los nuevos modos de transformada 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. Únicamente una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

1) Modo de transformada QH0: la TU0 superior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-4 y DCT-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-4 y DST-4, respectivamente.

2) Modo de transformada QH1: la TU1 superior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-4 y DCT-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-4 y DST-4, respectivamente.

3) Modo de transformada QH2: la TU1 inferior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-4 y DST-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-4 y DCT-4, respectivamente.

4) Modo de transformada QH3: la TU2 inferior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada,

en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-4 y DST-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-4 y DCT-4, respectivamente.

5 En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para QH0, QH1, QH2, y QH3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-4/DST-4, DST-4/DCT-4, DCT-4/DST-4, y DCT-4/DCT-4, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre QH0, QH1, QH2, y QH3.

10 En un ejemplo, los nuevos modos de transformada 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. Únicamente una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

20 1) Modo de transformada QV0: la TU0 superior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-8 y DCT-8, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-7 y DST-7, respectivamente.

25 2) Modo de transformada QV1: la TU1 superior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-7 y DCT-8, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-8 y DST-7, respectivamente.

30 3) Modo de transformada QV2: la TU1 inferior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-8 y DST-7, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-7 y DCT-8, respectivamente.

35 4) Modo de transformada QV3: la TU2 inferior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-7 y DST-7, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-8 y DCT-8, respectivamente.

40 En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para QV0, QV1, QV2, y QV3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-7/DST-7, DST-7/DCT-8, DCT-8/DST-7, y DCT-8/DCT-8, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre QV0, QV1, QV2, y QV3.

45 En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en un ejemplo, los nuevos modos de transformada 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. Únicamente una de las TU de tamaño $(W/2) \times (H/2)$ tiene residual. La TU con residual tiene un conjunto de tipos de transformada que incluye dos tipos de transformada, es decir un primer tipo de transformada y un segundo tipo de transformada, y los tipos de transformada son dependientes del modo de transformada. El tipo de transformada real usado por la TU con residual se señala en flujo de bits, por ejemplo, señalado por un marcador.

55 1) Modo de transformada QV0: la TU0 superior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DCT-4 y DCT-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU0 son DST-4 y DST-4, respectivamente.

60 2) Modo de transformada QV1: la TU1 superior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-4 y DCT-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-4 y DST-4, respectivamente.

65

3) Modo de transformada QV2: la TU1 inferior-izquierda tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DCT-4 y DST-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU1 son DST-4 y DCT-4, respectivamente.

4) Modo de transformada QV3: la TU2 inferior-derecha tiene residual; el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DST-4 y DST-4, respectivamente; un segundo tipo de transformada es que la transformada horizontal y la transformada vertical de TU2 son DCT-4 y DCT-4, respectivamente.

En otro ejemplo, el segundo tipo de transformada es siempre DCT-2/DCT-2 para QV0, QV1, QV2, y QV3.

En otro ejemplo, la TU con residual tiene un conjunto de tipos de transformada que incluye cuatro tipos de transformada, por ejemplo, DST-4/DST-4, DST-4/DCT-4, DCT-4/DST-4, y DCT-4/DCT-4, y los cuatro tipos de transformada son los mismos para cada modo de transformada entre QV0, QV1, QV2, y QV3.

En un ejemplo, los nuevos modos de transformada permitidos para CU incluyen 6 modos entre los modos mencionados anteriormente: HH0, HH1, HH2, VH0, VH1 y VH2. Se puede usar un marcador sector-nivel para señalar si se permiten los 6 modos de transformada para un sector particular. Donde los nuevos modos de transformada permitidos para CU significan que cuando se decodifica una CU, el objetivo de modo de transformada para la TU residual de la CU se puede obtener entre los nuevos modos de transformada permitidos para CU. Los nuevos modos de transformada permitidos para CU pueden ser un subconjunto de los 22 modos de transformada mencionados anteriormente, así puede reducir los modos de transformada candidatos para el objetivo de modo de transformada, para mejorar la eficiencia de decodificación, y reducir los bits necesarios para transmitir el índice/índices para indicar el objetivo de modo de transformada, para ahorrar el ancho de banda necesario para transmitir los datos de vídeo codificados. Se debe observar que, los nuevos modos de transformada permitidos para CU pueden ser predefinidos/preestablecidos o determinados en tiempo real, cuando los nuevos modos de transformada permitidos para CU son predefinidos/preestablecidos/determinados, el decodificador sabrá qué nuevos modos de transformada son modos de transformada candidatos para la TU residual de la CU.

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

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

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

En un ejemplo, los nuevos modos de transformada permitidos para CU incluyen 8 modos entre los modos mencionados anteriormente de transformada: HH0, HH1, VH0, VH1, HQ0, HQ1, VQ0 y VQ1.

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

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

En un ejemplo, los nuevos modos de transformada permitidos para una pluralidad de CU en un sector se señalan en conjunto de parámetros de secuencia (SPS) o la cabecera de sector del sector. Los modos HH0, HH1, HH2, VH0, VH1 y VH2 se agrupan como primer grupo. Los modos Q0, Q1, Q2, Q3, HQ0, HQ1, VQ0 y VQ1 se agrupan como segundo grupo. El permitir que cada grupo se señale por un marcador, es decir, un primer marcador se señala para indicar si se permite el primer grupo de modos, y un segundo marcador se señala para indicar si se permite el segundo grupo de modos.

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

Los nuevos modos de transformada pueden aplicarse a todas las CU o aplicarse las CU según la información de predicción de la CU. En un ejemplo, los nuevos modos de transformada se aplican a CU intrapredichas únicamente. En otro ejemplo, los nuevos modos de transformada se aplican a CU interpredichas únicamente. En otro ejemplo, los nuevos modos de transformada pueden permitirse a CU usando método de interpredicción específico (por ejemplo, compensación de movimiento basada en modelo de traslación) pero no se permite a CU que usan otro método de interpredicción (por ejemplo, compensación de movimiento basada en modelo afín). En otro ejemplo, los nuevos modos de transformada pueden permitirse para CU predichas por modo de fusión o por AMVP (predicción de vector de movimiento avanzada) modo con precisión de diferencia de vector de movimiento 1/4-pel, pero no se permite para CU que usan modo de fusión afín, intermodo afín o modo AMVP con precisión de diferencia de vector de movimiento 1-pel o 4-pel. En otro ejemplo, los nuevos modos de transformada pueden permitirse para CU que usan modo de fusión con un índice de fusión menor que 2, pero no se permiten para CU que usan modo de fusión con un índice de fusión no menor que 2. El modo de fusión y el modo AMVP se pueden hacer referencia en H.265/HEVC estándar. A modo de fusión afín e intermodo afín se les puede hacer referencia en códec JEM (Joint Exploration Model) de JVET (Joint Video Exploration Team).

Los nuevos modos de transformada pueden permitirse según el tamaño de la CU. En un ejemplo, si una CU se pone en mosaico por uno del modo de transformada resulta en una TU con un lado menor que un umbral (por ejemplo, 4, o 8, o 16 píxeles de luma etc.), no se permite el modo de transformada para la CU. En un ejemplo, si una CU se pone en mosaico por uno del modo de transformada resulta en una TU con un lado mayor que un umbral (por ejemplo, 32, o 64, o 128 píxeles de luma etc.), no se permite el modo de transformada para la CU. En un ejemplo, una CU con un lado (es decir, anchura o altura) mayor que un umbral (por ejemplo, 32 o 64 píxeles de luma) es no se permite usar los nuevos modos de transformada.

La Figura 12 ilustra un método de decodificación de una realización de la presente divulgación, incluye:

1201. analizar sintácticamente un flujo de bits recibido para obtener información de predicción de una CU; Este proceso se usa comúnmente en decodificación de vídeo, tal como analizar sintácticamente unidad de codificado y estructura de sintaxis de unidad de predicción en H.265/HEVC o H.266/VVC.
1202. obtener un objetivo de modo de transformada de la TU residual cuando la CU tiene únicamente una unidad de transformada residual (TU) y el tamaño de la TU residual es menor que el tamaño de la CU; en donde el objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual (por ejemplo, descrito por el índice de la TU residual entre todas las TU de la CU, o descrito por las coordenadas de la esquina superior izquierda de la TU residual y la anchura y la altura de la TU residual), y un tipo de transformada de la TU residual;
- El objetivo de modo de transformada 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 maneras de implementación, a fin de mejorar la eficiencia de decodificación, primero se comprueba la presencia de información de unidad de transformada en la CU. Donde La presencia de información de unidad de transformada puede indicarse por un elemento de sintaxis de cbf de raíz. Cbf de raíz igual a 0 indica la información de unidad de transformada de la CU no está presente en el flujo de bits (es decir, la CU no tiene residual), mientras cbf de raíz igual a 1 indica que la información de unidad de transformada de la CU está presente en el flujo de bits. Para una CU interpredicha, cbf de raíz igual a 1 implica que la CU tiene residual. La Etapa 1102 se realiza cuando la CU tiene información de unidad de transformada. Cuando la CU no tiene información de unidad de transformada, el residual de la CU se establece como cero y realizar etapa 1105 directamente.

1203. analizar sintácticamente el flujo de bits recibido para obtener coeficientes de transformada de la TU residual; Este proceso se usa comúnmente en decodificación de vídeo.
1204. aplicar una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados. Este proceso se usa comúnmente en decodificación de vídeo.
1205. Aplicar, en función del objetivo de modo de transformada, una transformada inversa a los coeficientes descuantizados para obtener un bloque residual de la TU residual. Este proceso se usa comúnmente en decodificación de vídeo. Además de la TU residual, los residuales de otras TU de la CU se establecen como cero.
1206. obtener un bloque de predicción de la CU en función de la información de predicción; Este proceso se usa comúnmente en decodificación de vídeo, tal como la intrapredicción y interpredicción en HEVC y VVC.
1207. obtener un bloque de vídeo en función del bloque residual y el bloque de predicción; Este proceso se usa comúnmente en decodificación de vídeo. En una manera de implementación, el bloque de vídeo se obtiene añadiendo el residual de la CU con la predicción de la CU.

1208. sacar una secuencia de vídeo, la secuencia de vídeo que incluye un fotograma de vídeo que incluye el bloque de vídeo.

Este proceso se usa comúnmente en decodificación de vídeo.

La Figura 12B ilustra otro método de decodificación de una realización de la presente divulgación.

La diferencia entre la realización que se muestra en la figura 12B y la realización que se muestra en la figura 12A reside en:

1202'. obtener un objetivo de modo de transformada de la TU residual cuando la CU tiene únicamente una unidad de transformada residual (TU) y el tamaño de la TU residual es menor que el tamaño de la CU; en donde el objetivo de modo de transformada especifica un modo de particionamiento 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 la anchura y la altura de la TU residual), y un conjunto de tipos de transformada de la TU residual; y obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada;

1205'. Aplicar, en función del tipo de transformada de la TU residual o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual, una transformada inversa a los coeficientes descuantizados para obtener un bloque residual de la TU residual. Las otras etapas hacen referencia a la realización que se muestra en la figura 12A, no se repiten aquí.

En una manera de implementación, obtener el objetivo de modo de transformada de la TU residual comprende: analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada de la TU residual; y obtener el objetivo de modo de transformada de la TU residual en función del índice de modo. Por ejemplo, si en un sector se permite usar 8 modos entre los 22 objetivos de modo de transformada mencionados anteriormente, el índice de modo puede indicarse por un índice de modo con M bins (por ejemplo, $M = 3, 4$ o 5), es decir, el índice de modo se binariza en M bins en proceso de Codificado Aritmético Binario Adaptativo basado en Contexto (CABAC) (donde M puede ser diferente para diferentes esquemas de binarización), o el índice de modo puede expresarse por M marcadores.

A fin de reducir el presupuesto de bits del índice de modo, en otra manera de implementación, antes de analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar los objetivos de modo de transformada de la TU residual, el método comprende además: determinar objetivos de modos de transformada candidatos de la TU residual en función de un tamaño de la CU; donde analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar los objetivos de modo de transformada de la TU residual comprende: analizar sintácticamente, en función de los objetivos de modos de transformada candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar los objetivos de modo de transformada de la TU residual. Por lo tanto, dependiendo del tamaño de la CU, la cantidad de los objetivos de modos de transformada candidatos puede ser menor que 8, así 1 o más bins del modo marcador puede guardarse.

Los objetivos de modos de transformada candidatos pueden basarse en el tamaño de CU. Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3.

Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada VH0 y modo de transformada VH1;

Por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th3]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos pueden incluir modo de transformada HQ0, modo de transformada HQ1 y modo de transformada HH2;

Por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos pueden incluir modo de transformada HH0 y modo de transformada HH1;

Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

Por ejemplo, cuando una anchura 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 objetivos de modos de transformada candidatos pueden incluir modo de transformada VH0 y modo de transformada VH1;

Por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 4, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos pueden incluir modo de transformada HQ0, modo de transformada HQ1 y modo de transformada HH2; o

Por ejemplo, cuando una altura de la CU está en el intervalo de $[Th1 \times 2, Th2]$ y una anchura de la CU está en el intervalo de $[Th1, Th2]$, los objetivos de modos de transformada candidatos pueden incluir modo de transformada HH0 y modo de transformada HH1.

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 manera de implementación, a fin de reducir el presupuesto de bits para indicar el objetivo de modo de transformada, los 22 modos mencionados anteriormente se agrupan en grupos de modos. Por ejemplo, los grupos de modos pueden incluir al menos un grupo según lo siguiente:

un grupo de modos incluye modo de transformada HH0, modo de transformada HH1, modo de transformada HH2, modo de transformada VH0, modo de transformada VH1 y modo de transformada VH2;

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

un grupo de modos incluye modo de transformada HH0, HH1, HH2, VH0, VH1 y VH2;

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

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

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

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

un grupo de modos incluye modo de transformada Q0, modo de transformada Q1, modo de transformada Q2, modo de transformada Q3, modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0, modo de transformada VQ1, modo de transformada HH0, modo de transformada HH1, modo de transformada HH2, modo de transformada VH0, modo de transformada VH1 y modo de transformada VH2;

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

un grupo de modos incluye modo de transformada HH0, modo de transformada HH1, modo de transformada VH0 y modo de transformada VH1;

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

un grupo de modos incluye modo de transformada HQ0, modo de transformada HQ1, modo de transformada VQ0 y modo de transformada VQ1;

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

En una manera de implementación, antes de analizar sintácticamente el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada, el método comprende además: determinar al menos un candidato de grupos de modos de la TU residual en función de un tamaño de la CU; y correspondientemente, analizar sintácticamente, en función del al menos un candidato de grupos de modos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada.

Por ejemplo, cuando una anchura 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 modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

Por ejemplo, cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

Por ejemplo, cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

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

Por ejemplo, cuando una anchura 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 modo de transformada Q0, modo de transformada Q1, modo de transformada Q2 y modo de transformada Q3;

Por ejemplo, cuando una anchura 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 modo de transformada VQ0, modo de transformada VQ1 y modo de transformada VH2;

Por ejemplo, cuando una anchura 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 modo de transformada VH0 y modo de transformada VH1;

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

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

El umbral $Th1$, $Th2$ 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.

En otra manera de implementación, antes de obtener el objetivo de modo de transformada de la TU residual, el método comprende además: determinar, en función de la información de predicción de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Por ejemplo, cuando un modo de predicción de la CU es modo de intrapredicción, determinar obtener el objetivo de modo de transformada de la TU residual;

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

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

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

En otra manera de implementación, antes de obtener el objetivo de modo de transformada de la TU residual, el método comprende además: determinar, en función de un tamaño de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Por ejemplo, cuando una anchura o una altura de la CU es mayor que un umbral de CU preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual. El umbral de CU preestablecido puede ser 32 píxeles de luma, 64 píxeles de luma o 128 píxeles de luma.

Por ejemplo, cuando una anchura o una altura de una TU de la CU es menor que un umbral de TU mínimo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual. El umbral de TU mínimo preestablecido puede ser 4 píxeles de luma, 8 píxeles de luma o 16 píxeles de luma.

- 5 Por ejemplo, cuando una anchura o una altura de una TU de la CU es mayor que un umbral de TU máximo preestablecido, determinar no obtener el objetivo de modo de transformada de la TU residual. El umbral de TU máximo preestablecido es 32 píxeles de luma, 64 píxeles de luma o 128 píxeles de luma.

10 Se debe observar que, incluso según la información de predicción de la CU y/o el tamaño de la CU, determinar obtener el objetivo de modo de transformada de la TU residual, únicamente significa que el objetivo de modo de transformada puede obtenerse, pero no significa que ciertamente se va a obtener el objetivo de modo de transformada. Como en el lado de codificador, elegir qué modo de transformada para hacer la transformación depende de muchos factores. Esto es, los nuevos modos de transformada mencionados anteriormente (los 22 modos de transformada mencionados anteriormente) se añaden como candidato, los modos de transformada ya existentes (modos de transformada antiguos), por ejemplo, la CU se transforma por una TU del tamaño de CU, todavía se puede usar.

15 Por lo tanto, sobre la base de la CU que tiene información de unidad de transformada, cuando se determina, según la información de predicción de la CU y/o el tamaño de la CU, obtener el objetivo de modo de transformada de la TU residual; analizar sintácticamente el flujo de bits puede obtener un modo marcador que indica los modos de transformada antiguos. Por consiguiente, cuando se determina no obtener el objetivo de modo de transformada de la TU residual, que significa que los modos de transformada antiguos se usan para la CU. Si la CU se transforma por una TU del tamaño de CU, la TU residual es del tamaño de CU, y el tipo de transformada de la TU residual puede ser DCT-2 para transformada horizontal y DCT-2 para transformada vertical de manera predeterminada o puede señalarse en flujo de bits (por ejemplo, el método de AMT en software VTM, en el que se permiten múltiples tipos de transformada para una TU y el tipo de transformada usado por la TU se señala).

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

25 Si la CU usa uno de los objetivos de modos de transformada candidatos, analizar sintácticamente el flujo de bits para obtener el modo de transformada 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 transformada de la CU, analizando sintácticamente un índice de grupo, y entonces decide el objetivo de modo de transformada dentro de grupo analizando sintácticamente otro índice.

30 Analizar sintácticamente el índice de grupo puede basarse en qué grupos de modos se determinan/prestablecen como grupos de modos candidatos. Por ejemplo, cuando el grupo A y cualquier otro grupo son grupos de modos candidatos, un primer marcador se analiza sintácticamente para determinar si el objetivo de modo de transformada pertenece al grupo A o no; de otro modo, si el primer marcador no está presente en el flujo de bits, y se infiere que el modo de transformada pertenece al grupo A si grupo A es un candidato grupo de modos, o se infiere que pertenece a los grupos B, C, D y E si el grupo A no es un candidato grupo de modos. Si el objetivo de modo de transformada pertenece a los grupos B, C, D y E, un segundo marcador se analiza sintácticamente para determinar si el objetivo de modo de transformada 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 marcador se puede inferir si únicamente B/D es candidato o si únicamente C/E es candidato. Si el modo de transformada pertenece a B/D, un tercer marcador puede analizarse sintácticamente para determinar si el objetivo de modo de transformada pertenece a B. De manera similar, si el objetivo de modo de transformada pertenece a C/E, el tercer marcador puede analizarse sintácticamente para determinar si el modo de transformada pertenece a C.

35 Similar a analizar sintácticamente el índice de grupo, analizar sintácticamente el índice del objetivo de modo de transformada dentro de un grupo puede basarse en qué modos de transformada están en el grupo.

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

45 Si la CU usa uno de los objetivos de modos de transformada candidatos, analizar sintácticamente el flujo de bits para obtener el modo de transformada de la CU. En un método, modos de transformada 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 que al que pertenece el objetivo de modo de transformada, analizando sintácticamente un índice de grupo, y entonces decide el objetivo de modo de transformada dentro de grupo analizando sintácticamente un índice de modo.

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

- 5 una unidad de decodificación por entropía 304, se configura para analizar sintácticamente un flujo de bits recibido para obtener información de predicción de una unidad de codificado (CU); obtener un objetivo de modo de transformada de una unidad de transformada residual (TU) 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 objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual, y un conjunto de tipos de transformada de la TU residual; obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada; y analizar sintácticamente el flujo de bits recibido para obtener coeficientes de transformada de la TU residual. Donde el objetivo de modo de transformada es el mismo que se ha mencionado anteriormente.
- 10 una unidad de procesamiento de cuantización inversa 310, se configura para aplicar una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados;
- 15 una unidad de procesamiento de transformada inversa 312, se configura para aplicar una transformada inversa a los coeficientes descuantizados en función del tipo de transformada de la TU residual o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual para obtener un bloque residual de la TU residual;
- 20 una unidad de procesamiento de predicción 360, se configura para obtener un bloque de predicción de la CU en función de la información de predicción;
- una unidad de reconstrucción 314, se configura para obtener un bloque de vídeo en función del bloque residual y el bloque de predicción; y
- 25 un salida 332, se configura para tener como salida una secuencia de vídeo, la secuencia de vídeo incluye un fotograma de vídeo que incluye el bloque de vídeo.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada de la TU residual; y obtener el objetivo de modo de transformada de la TU residual en función del índice de modo.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para determinar objetivos de modos de transformada candidatos de la TU residual en función de un tamaño de la CU; y analizar sintácticamente, en función de los objetivos de modos de transformada candidatos de la TU residual, el flujo de bits para obtener un índice de modo para indicar los objetivos de modo de transformada de la TU residual.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para determinar objetivos de modos de transformada candidatos de la TU residual según el método como se ha indicado anteriormente.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para analizar sintácticamente el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada; y analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada en el grupo de modos; y obtener el objetivo de modo de transformada en función del índice de grupo y el índice de modo.

Donde el objetivo de modo de transformada pertenece a cualquier grupo de modos como se ha indicado anteriormente.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para determinar al menos un candidato de grupos de modos de la TU residual en función de un tamaño de la CU; y analizar sintácticamente, en función del al menos un candidato de grupos de modos de la TU residual, el flujo de bits para obtener un índice de grupo para indicar a qué grupo de modos pertenece el objetivo de modo de transformada.

Donde la unidad de decodificación por entropía se configura además para determinar al menos un candidato de grupos de modos según al menos un del método como se ha indicado anteriormente.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para determinar, en función de la información de predicción de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Donde la unidad de decodificación por entropía 304, se configura además para determinar si obtener el objetivo de modo de transformada de la TU residual según al menos uno de los métodos como se ha indicado anteriormente.

En una manera de implementación, la unidad de decodificación por entropía 304 se configura además para determinar, en función de un tamaño de la CU, si obtener el objetivo de modo de transformada de la TU residual.

Donde la unidad de decodificación por entropía 304, se configura además para determinar si obtener el objetivo de modo de transformada de la TU residual según al menos uno de los métodos como se ha indicado anteriormente.

La presente divulgación divulga además un decodificador que comprende circuitería de procesamiento para llevar a cabo el método como se ha indicado anteriormente.

5 La presente divulgación divulga además un producto de programa informático que comprende un código de programa para realizar el método como se ha indicado anteriormente.

La presente divulgación divulga además un decodificador, que comprende:

10 uno o más procesadores; y
un soporte de almacenamiento no transitorio legible por ordenador acoplado a los procesadores y que almacena programación para ejecución por los procesadores, en donde la programación, cuando es ejecutada por los procesadores, configura el decodificador para llevar a cabo el método como se ha indicado anteriormente.

15 La FIGURA 13 ilustra un ejemplo de intrapredicción 1300 en codificado de vídeo que emplea modos de intrapredicción. Como se muestra, un bloque actual 1301 se puede predecir por muestras en bloques vecinos 1310. Los codificadores generalmente pueden codificar una imagen de parte superior izquierda a parte inferior derecha. Sin embargo, los codificadores pueden codificar de derecha a izquierda en algunos casos como se discute más adelante. Se debe observar que, como se emplea en esta memoria derecha se refiere al lado derecho de una imagen codificada, izquierda se refiere al lado izquierdo de una imagen codificada, parte superior se refiere al lado superior de una imagen codificada, y parte inferior se refiere al lado inferior de una imagen codificada.

25 Se debe observar que un bloque actual 1301 puede no siempre coincidir exactamente con una muestra de un bloque vecino 1310. En tal caso, un modo de predicción se codifica del bloque vecino 1310 coincidente más cercano. A fin de permitir al decodificador determinar el valor apropiado, se retiene una diferencia entre el valor predicho y el valor real. Esto se denomina información de residual. La información de residual ocurre tanto en intrapredicción 1300 así como en interpredicción.

30 La FIGURA 14 es un diagrama esquemático de un ejemplo de mecanismo de codificación de vídeo 1400, basado en intrapredicción 1300 y/o interpredicción. Un bloque de imagen 1401 se puede obtener por un codificador de uno o más fotogramas. Por ejemplo, una imagen se puede partir en una pluralidad de regiones de imagen rectangulares. Cada región de la imagen corresponde a una unidad de árbol de codificado (CTU). Una CTU se particiona en una pluralidad de bloques, tales como las unidades de codificado en HEVC. La información de partición de bloque se codifica entonces en un flujo de bits 1411. Por consiguiente, el bloque de imagen 301 es una parte particionada de una imagen y contiene píxeles que representan componentes luma y/o componentes croma en una parte correspondiente de la imagen. Durante la codificación, el bloque de imagen 1401 se codifica como bloque de predicción 303 que contiene información de predicción tal como modos de predicción para intrapredicción y/o vectores de movimiento para interpredicción. Codificar el bloque de imagen 1401 como bloque de predicción 1403 puede entonces dejar un bloque residual 1405 que contiene información de residual que indica la diferencia entre el bloque de predicción 1403 y el bloque de imagen 1401.

40 Se debe observar que un bloque de imagen 1401 puede partitionarse como unidad de codificado que contiene un bloque de predicción 1403 y un bloque residual 1405. El bloque de predicción 1403 puede contener todas las muestras de predicción de la unidad de codificado, y el bloque residual 1405 puede contener todas las muestras residuales de la unidad de codificado. En tal caso, el bloque de predicción 1403 es del mismo tamaño que el bloque residual 1405. En otro ejemplo, el bloque de imagen 1401 puede partitionarse como unidad de codificado que contiene dos bloques de predicción 1403 y un bloque residual 1405. En tal caso, cada bloque de predicción 1403 contiene una parte de las muestras de predicción de la unidad de codificado, y el bloque residual 1405 contiene todas de las muestras residuales de la unidad de codificado. En incluso otro ejemplo, el bloque de imagen 1401 se particiona en una unidad de codificado que contiene dos bloques de predicción 1403 y cuatro bloques residuales 1405. El patrón de partición de los bloques residuales 1405 en una unidad de codificado puede señalarse en el flujo de bits 1411. Tales patrones de posición pueden incluir Árbol Cuádruple de Residual (RQT) en HEVC. Además, un bloque de imagen 1401 puede contener únicamente componentes luma (por ejemplo, luz), denotada como componentes Y, de muestras de imágenes (o píxeles). En otros casos, el bloque de imagen 1401 puede contener componentes Y, U y V de muestras de imágenes, donde U y V indican componentes de crominancia (por ejemplo, color) en un espacio de color de luminancia azul y luminancia roja (UV).

55 Puede emplearse SVT para comprimir aún más la información. Específicamente, SVT emplea un bloque de transformada 1407 para comprimir aún más el bloque residual 1405. El bloque de transformada 1407 contiene una transformada, tal como DCT inversa y/o DST inversa. La diferencia entre el bloque de predicción 1403 y el bloque de imagen 1401 es el ajuste a la transformada empleando coeficientes de transformada. Al indicar el modo de transformada del bloque de transformada 1407 (por ejemplo, DCT inversa y/o DST inversa) y los correspondientes coeficientes de transformada, el decodificador puede reconstruir el bloque residual 1405. Cuando no se requiere reproducción exacta, los coeficientes de transformada se pueden comprimir aún más redondeando ciertos valores para crear un mejor ajuste para la transformada. Este proceso se conoce como cuantización y se realiza según parámetros de cuantización que describen la cuantización permisible. Por consiguiente, los modos de transformada,

los coeficientes de transformada y los parámetros de cuantización del bloque de transformada 1407 se almacenan como información de residual de transformado en un bloque residual transformado 1409, que también se pueden denominar simplemente bloque residual en algunos casos.

La información de predicción del bloque de predicción 1403 y la información de residual transformada del bloque residual transformado 1409 puede 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 de imagen 1401. Específicamente, el decodificador puede emplear la información de residual transformada para determinar el bloque de transformada 1407. El bloque de transformada 1407 puede entonces emplearse conjuntamente con el bloque residual transformado 1409 para determinar el bloque residual 1405. El bloque residual 1405 y el bloque de predicción 1403 pueden entonces emplearse para reconstruir el bloque de imagen 1401. El bloque de imagen 1401 puede entonces posicionarse respecto a otros bloques de imagen decodificada 1401 para reconstruir fotogramas y posicionar tales fotogramas para recuperar el vídeo codificado.

Ahora se describe SVT más en detalle. Para realizar SVT, el bloque de transformada 1407 se selecciona para ser menor que el bloque residual 1405. El bloque de transformada 1407 se emplea para transformar una parte correspondiente del bloque residual 1405, y dejar el resto del bloque residual sin codificado/compresión adicional. Esto es porque la información de residual generalmente no se distribuye uniformemente por el bloque residual 1405. SVT emplea un bloque de transformada más pequeño 1407 con una posición adaptativa para capturar la mayoría de la información de residual en el bloque residual 1405, sin que se necesite transformar el bloque residual 1405 entero. Este planteamiento puede lograr mejor eficiencia de codificado que transformar toda la información de residual en el bloque residual 1405. Como el bloque de transformada 1407 es menor que el bloque residual 1405, SVT emplea un mecanismo para señalar la posición de la transformada respecto al bloque residual 1405. Por ejemplo, cuando se aplica SVT para un bloque residual 1405 de tamaño $w \times h$ (por ejemplo, anchura por altura), el tamaño y la información de posición del bloque de transformada 1407 puede codificarse en el flujo de bits 1411. Esto permite al decodificador reconstruir el bloque de transformada 1407 y componer el bloque de transformada 1407 en la posición correcta respecto al bloque residual transformado 1409 para reconstrucción del bloque residual 1405.

Se debe observar que algunos bloques de predicción 1403 se pueden codificar sin dar como resultado un bloque residual 1405. Sin embargo, este tipo de caso no da como resultado el uso de SVT y por tanto no se discute más. Como se señala anteriormente, SVT puede emplearse para bloques interpredichos o bloques intrapredichos. Además, SVT puede emplearse en bloques residuales 1405 generados por mecanismos de interpredicción especificados (por ejemplo, compensación de movimiento basada en modelo de traslación), pero puede no emplearse para bloques residuales 1405 generados por otros mecanismos de interpredicción especificados (por ejemplo, compensación de movimiento basada en modelo afín).

La FIGURA 15 ilustra ejemplos de transformadas SVT 1500 que pueden emplearse como bloque de transformada para codificar un bloque residual. Las transformadas SVT 1500 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 denotan la anchura y la altura del bloque de transformada, respectivamente, y w , h denotan la anchura y la altura del bloque residual, respectivamente. Por ejemplo, la anchura y la altura del bloque de transformada son ambos la mitad de la anchura y la altura del bloque residual. SVT-II se describe como $w_t = w/4$, $h_t = h$ donde las variables son como se ha descrito anteriormente. Por ejemplo, la anchura de bloque de transformada es un cuarto de la anchura del bloque residual y la altura del bloque de transformada 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 ha descrito anteriormente. Por ejemplo, la anchura del bloque de transformada es igual a la anchura del bloque residual y la altura del bloque de transformada es un cuarto de la altura del bloque residual. La información de tipo que indica el tipo de SVT bloque (por ejemplo, SVT-I, SVT-II, o SVT-III) se codifica en flujo de bits para soportar reconstrucción por el decodificador.

Como se puede ver con la FIGURA 15, cada transformada se puede posicionar en diversas ubicaciones respecto al bloque residual. La posición del bloque de transformada se representa por un desplazamiento de posición (x, y) , a la esquina superior izquierda del bloque residual, donde x indica la distancia horizontal entre la esquina superior izquierda del bloque de transformada y la del bloque residual en la unidad de píxeles, e y indica la distancia vertical entre la esquina superior izquierda del bloque de transformada y la del bloque residual en la unidad de píxeles. Cada posición potencial del bloque de transformada dentro del bloque residual se denomina una 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 flujo de bits junto con el tipo de SVT bloque empleado. Para reducir la complejidad para SVT-I, un subconjunto de treinta y dos posiciones se puede seleccionar de las ochenta y una posibles posiciones candidatas. Este subconjunto entonces actúa como las posiciones candidatas permitidas para SVT-I.

Un inconveniente de un esquema de SVT que emplea transformadas SVT 1500 es que codificar la información de posición de SVT como información de residual resulta en una significativa sobrecarga de señalización. Adicionalmente, la complejidad de codificador puede aumentarse significativamente conforme aumenta el número de posiciones probado por procesos de calidad de compresión, tales como optimización de distorsión de tasa (RDO). Puesto que el

número de posiciones candidatas aumenta con el tamaño del bloque residual, la sobrecarga de señalización se puede incluso mayor para bloques residuales más grandes, tales como 32×32 o 64×128 . Otro inconveniente de emplear transformadas SVT 1500 es que el tamaño del bloque de transformada es un cuarto del tamaño del bloque residual. Un bloque de transformada de este tipo de tamaño puede no ser suficientemente grande como para cubrir la mayor información de residual en el bloque residual en muchos casos.

La FIGURA 16 ilustra ejemplos adicionales de transformadas SVT 1600 que pueden emplearse como bloque de transformada 307 para codificar un bloque residual 305. Las transformadas SVT 1600 se denominan SVT vertical (SVT-V) y SVT horizontal (SVT-H). Las transformadas SVT 1600 son similares a las transformadas SVT 1500, pero se diseñan para soportar 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 como se ha descrito anteriormente. La anchura del bloque de transformada es la mitad de la anchura del bloque residual y el bloque de transformada de altura 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 ha descrito anteriormente. Por ejemplo, la anchura del bloque de transformada es igual a la anchura del bloque residual y el bloque de transformada de altura 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 transformada en SVT-V y SVT-H se agranda a la mitad del bloque residual, que puede cubrir más información de residual en el bloque residual.

Como con transformadas SVT 1500, las transformadas SVT 1600 pueden incluir varias posiciones candidatas, donde las posiciones candidatas son la posible posición permisible de un bloque de transformada respecto a un bloque residual. Las posiciones candidatas se determinan según un tamaño de etapa de posición candidata (CPSS). Las posiciones candidatas pueden 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 como la posición seleccionada para una transformada se puede señalar con menos bits. Además, reducir el número de posiciones candidatas hace algorítmicamente más simple la selección de una posición de transformada, lo que permite reducir la complejidad de codificador (por ejemplo, dando como resultado que se empleen menos recursos de cómputo para codificar).

La FIGURA 17 ilustra ejemplos de posiciones candidatas de transformada SVT 1700 respecto a un bloque residual. Específicamente, las posiciones candidatas de transformada SVT 1700 emplean SVT-V y SVT-H (por ejemplo, transformadas SVT 1600) posicionadas respecto a un bloque residual. Las posiciones candidatas permisibles para una transformada dependen del CPSS, que además depende de la parte del bloque residual que el bloque de transformada debe cubrir y/o un tamaño de etapa entre posiciones candidatas. Por ejemplo, el CPSS puede calcularse como $s = w/M1$ para SVT-V o $s = h/M2$ para SVT-H, donde w y h son la anchura y la altura del bloque residual, respectivamente, y $M1$ y $M2$ son enteros predeterminados en el intervalo de dos a ocho. Se permiten más posiciones candidatas con un valor más grande de $M1$ o $M2$. Por ejemplo, $M1$ y $M2$ pueden establecerse ambos como ocho. En este caso, un valor del índice de posición (P) que describe la ubicación del bloque de transformada 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 enteros predefinidos que especifican un tamaño de etapa mínimo. $Th1$ y $Th2$ pueden ser enteros que no son menores que dos. En este ejemplo, $Th1$ y $Th2$ se establecen como cuatro, y $M1$ y $M2$ se establecen como ocho. En este ejemplo, diferentes tamaños de bloque pueden tener un número diferente de posiciones candidatas. Por ejemplo, cuando la anchura del bloque residual es ocho, hay disponibles dos posiciones candidatas para SVT-V, específicamente posiciones candidatas 1700 (a) y (e). Como ejemplo únicamente dos posiciones candidatas satisfacen el CPSS cuando el tamaño de etapa, como se indica por $Th1$, es grande y la parte del bloque residual la transformada cubre, como se indica por $w/M1$, es también grande. Sin embargo, cuando w se establece a dieciséis, la parte del bloque residual que la transformada cubre disminuye debido a un cambio en $w/M1$. Esto resulta en más posiciones candidatas, en este caso las tres posiciones candidatas 1700 (a), (c), y (e). Las cinco posiciones candidatas están disponibles cuando la anchura del bloque residual es mayor que dieciséis mientras los valores de $Th1$ y $M1$ son como se ha discutido anteriormente.

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

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 ha discutido anteriormente. En este caso, $T1$ y $T2$ se establecen como 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 la transformada cubre depende

de si la altura del bloque residual es mayor que la anchura del bloque residual, o viceversa. Por consiguiente, el número de posiciones candidatas para SVT-H o SVT-V además depende 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 ha discutido 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 sintácticamente de un elemento de sintaxis, y Th1 y Th2 puede compartir el mismo valor analizado sintácticamente de otro elemento de sintaxis.

La FIGURA 18 ilustra ejemplos de posiciones de transformada SVT 1800 respecto a un bloque residual. Las posiciones de transformada SVT 1800 se seleccionan de posiciones candidatas, tales como posiciones candidatas de transformada SVT 1700. Específicamente, las posiciones de transformada SVT 1800 seleccionadas pueden codificarse en cuanto a un índice de posición P. El índice de posición P se puede emplear para determinar un desplazamiento de posición (Z) de la esquina superior izquierda del bloque de transformada 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 transformada en función de tipo de SVT y se calcula como se discute con

$$\frac{w-w_1}{s}$$

respecto a la FIG 6. El valor de P puede codificarse como 0, 1..., S cuando el bloque de transformada es un tipo

$$h-h_1$$

SVT-V. El valor de P puede codificarse como 0, 1..., S cuando el bloque de transformada es un tipo 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 transformada es (Z, 0) para SVT-V o (0, Z) para SVT-H.

Como se trata con mayor detalle más adelante, el codificador puede codificar el tipo de SVT de transformada (por ejemplo, SVT-H o SVT-T) y tamaño de bloque residual en el flujo de bits empleando marcadores. El decodificador puede entonces determinar el tamaño de transformada SVT en función del tamaño de transformada SVT y el tamaño de bloque residual. Una vez se determina el tamaño de transformada SVT, el decodificador puede determinar las posiciones candidatas permisibles de la transformada SVT, tales como posiciones candidatas de transformada SVT 1700, según la función CPSS. Como el decodificador es capaz de determinar las posiciones candidatas de la transformada SVT, el codificador puede no señalar las coordenadas del desplazamiento de posición. En cambio, se puede emplear un código para indicar cuál de las posiciones candidatas se emplea para la correspondiente transformada. Por ejemplo, el índice de posición P puede binarizarse en uno o más bins usando código unario truncado para aumentar la compresión. Como ejemplo particular, cuando el valor P está en el intervalo de cero a cuatro, el valor P 0, 4, 2, 3 y 1 se puede binarizar como 0, 01, 001, 0001 y 0000, respectivamente. Este código binario se comprime más que expresando el valor en base diez del índice de posición. Como otro ejemplo, cuando el valor P está en el intervalo de cero a uno, el valor P 0 y 1 se puede binarizar como 0 y 1, respectivamente. Como tal, el índice de posición puede crecer o contraerse en tamaño según se desee para señalar una posición de bloque de transformada particular a la luz de las posibles posiciones candidatas para el bloque de transformada.

El índice de posición P puede binarizarse en uno o más bins empleando la posición más probable y posiciones restantes menos probables. Por ejemplo, cuando los bloques vecinos izquierdo y superior ya se han decodificado en el decodificador, y por tanto están disponibles para 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, el valor 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, el valor P 2, 0 y 1 se binarizan como 1, 01 y 00, respectivamente. Por consiguiente, el índice de posición más probable de las posiciones candidatas se denota con los menos bits para reducir sobrecarga de señalización para los casos más comunes. La probabilidad se puede determinar en función del orden de codificado de bloques reconstruidos adyacentes. Por consiguiente, el decodificador puede inferir el esquema de palabra código a emplearse para un correspondiente bloque en función del esquema de decodificación empleado.

Por ejemplo, en HEVC la unidad de codificado orden de codificado es generalmente de parte superior a parte inferior y de izquierda a derecha. En tal caso, el lado derecho de una unidad de codificado de codificación/decodificación actual no está disponible que representa la parte esquina superior derecha la posición de transformada más probable. Sin embargo, se derivan predictores de vector de movimiento de los vecinos espaciales izquierdo y superior. En tal caso, la información de 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 una orden de unidad de codificado adaptativo de codificado, un nodo puede partir verticalmente en dos nodos subordinado y el nodo subordinado derecho puede codificarse antes que el izquierdo. En este caso, el vecino de lado derecho de la izquierda nodo subordinado se ha reconstruido antes de la decodificación/codificación de la nodo subordinado

izquierdo. Además, en tal caso, los píxeles vecinos de lado izquierdo son están disponibles. Cuando el vecino de lado derecho está disponible y el vecino de lado izquierdo no está disponible, la parte inferior izquierda del bloque residual es probable que contenga un gran cantidad de información de residual, y así la posición candidata que cubre la parte inferior izquierda del bloque residual se vuelve la posición más probable.

Por lo tanto, el índice de posición P puede binarizarse en uno o más bins según si el lado derecho junto al bloque residual se ha reconstruido. En un ejemplo, el valor P está en el intervalo de cero a 2, como se muestra por posiciones de transformada SVT 1800. Cuando el lado derecho junto al bloque residual se ha reconstruido, el valor P 0, 2 y 1 se binarizan como 0, 01 y 00. De otro modo, el valor P 2, 0 y 1 se binarizan como 0, 01 y 00. En otro ejemplo, cuando el lado derecho junto al bloque residual se ha reconstruido, pero el lado izquierdo junto al bloque residual no se ha reconstruido, el valor P 0, 2 y 1 se binarizan como 0, 00 y 01. De otro modo, el valor P 2, 0 y 1 se binarizan como 0, 00 y 01. En estos ejemplos, la posición correspondiente a un único bin es la posición más probable y las otras dos posiciones son posiciones restantes. Por ejemplo, la posición más probable es dependiente de la disponibilidad del vecino de lado derecho.

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

En otro ejemplo, cuando el bloque residual se asocia con interpredicción basada en coincidencia con plantilla, la posición más probable (por ejemplo, posición dos, o posición cero cuando el vecino derecho está disponible pero el vecino izquierdo no está disponible) se establece directamente como la posición de bloque de transformada e información de posición no se señala en el flujo de bits. De otro modo, el índice de posición se señala explícitamente en el flujo de bits.

También cabe señalar que se pueden emplear diferentes transformadas dependiendo de la posición del bloque de transformada respecto al bloque residual. Por ejemplo, el lado izquierdo del bloque residual se reconstruye y el lado derecho del bloque residual no se reconstruye, que ocurre para codificado de vídeo con un fijado unidad de codificado orden de codificado de izquierda a derecha y de parte superior a parte inferior (por ejemplo, el orden de codificado en HEVC). En este caso, la posición candidata que cubre la esquina inferior derecha del bloque residual puede emplear DST (por ejemplo, tipo de DST siete (DST-7) o tipo de DST un (DST-1)) para una transformada en un bloque de transformada cuando codificación. (En la manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en este caso, la posición candidata que cubre la esquina inferior derecha del bloque residual puede emplear DST (por ejemplo, tipo de DST siete (DST-4) o tipo de DST un (DST-1)) para una transformada en un bloque de transformada cuando codificación). Por consiguiente, una transformada se emplea DST inversa en el decodificador para la correspondiente posición candidata. Además, una posición candidata que cubre la esquina superior izquierda del bloque residual puede emplear DCT (por ejemplo, tipo de DCT ocho (DCT-8) o tipo de DCT dos (DCT-2)) para una transformada en el bloque de transformada cuando codificación. (En la manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, además, una posición candidata que cubre la esquina superior izquierda del bloque residual puede emplear DCT (por ejemplo, tipo de DCT ocho (DCT-4) o tipo de DCT dos (DCT-2)) para una transformada en el bloque de transformada cuando codificación). Por consiguiente, una transformada DCT inversa se emplea en el decodificador para la correspondiente posición candidata. Esto es porque la esquina inferior derecha es la más distante a el área espacialmente reconstruida entre las cuatro esquinas en este caso. Además, DST es más eficaz que DCT para transformar la información de residual distribución cuando el bloque de transformada cubre la esquina inferior derecha del bloque residual. Sin embargo, DCT es más eficaz que DST para transformar la información de residual distribución cuando el bloque de transformada cubre la esquina superior izquierda del bloque residual. Para las posiciones candidatas restantes, el tipo de transformada puede ser DST o DCT inversas. Por ejemplo, cuando una posición candidata está más cerca de la esquina inferior derecha que la esquina superior izquierda, se emplea DST inversa como el tipo de transformada. De otro modo DCT inversa se emplea el tipo de transformada.

Como ejemplo específico, pueden permitirse tres posiciones candidatas para la bloque de transformada 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 a está en el medio del bloque residual y es equidistante a ambas esquinas izquierda y derecha. Los tipos de transformada se pueden seleccionar como DCT-8, DST-7 y DST-7 para posición cero, posición uno y posición dos, respectivamente, en el codificador. Las transformadas inversas DCT-8, DST-7 y DST-7 pueden entonces emplearse en el decodificador para posición cero, posición uno y posición dos, respectivamente. En otro ejemplo, los tipos de transformada para posición cero, posición uno y posición dos son DCT-2, DCT-2 y DST-

7, respectivamente, en el codificador. Las transformadas inversas DCT-2, DCT-2 y DST-7 pueden entonces emplearse en el decodificador para posición cero, posición uno y posición dos, respectivamente. Como tal, se pueden predeterminar los tipos de transformada para correspondientes posiciones candidatas.

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, como ejemplo específico, pueden permitirse tres posiciones candidatas para el bloque de transformada 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 a está en el medio del bloque residual y es equidistante a ambas esquinas izquierda y derecha. Los tipos de transformada se pueden seleccionar como DCT-4, DST-4 y DST-4 para posición cero, posición uno y posición dos, respectivamente, en el codificador. Las transformadas inversas DCT-4, DST-4 y DST-4 pueden entonces emplearse en el decodificador para posición cero, posición uno y posición dos, respectivamente. En otro ejemplo, los tipos de transformada para posición cero, posición uno y posición dos son DCT-2, DCT-2 y DST-4, respectivamente, en el codificador. Las transformadas inversas DCT-2, DCT-2 y DST-4 pueden entonces emplearse en el decodificador para posición cero, posición uno y posición dos, respectivamente. Como tal, se pueden predeterminar los tipos de transformada para correspondientes posiciones candidatas.

En algunos casos, las múltiples transformadas dependientes de la posición descritas anteriormente pueden aplicarse a bloques de transformada luma únicamente. Los correspondientes bloques croma de transformada siempre pueden usar DCT-2 inversa en el proceso de transformada/transformada inversa.

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

En un ejemplo, el volteo horizontal ocurre cuando el bloque de transformada emplea DST/DST inversa en el proceso de transformada. En este caso, el vecino del lado derecho del bloque residual se codifica/reconstruye antes que el bloque actual, y el vecino del lado izquierdo no se codifica/reconstruye antes que el 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 la anchura del bloque de transformada y $i=0,1,\dots,(w/2)-1$. El volteo horizontal de las muestras residuales puede aumentar la eficiencia de codificado la hacer que la distribución residual encaje mejor a una transformada DST.

La FIGURA 20 es un diagrama de flujo de un ejemplo de método 2000 de decodificación de vídeo con SVT dependiente de posición que emplea la mecanismos discutido anteriormente. El método 2000 puede ser iniciado 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 bloques de predicción y bloques residuales transformados, tal como el bloque de predicción 303 y el bloque residual transformado 309. El método 2000 también determina bloques de transformada, tal como el bloque de transformada 307, que se emplea para determinar bloques residuales, tal como el bloque residual 305. Los bloques residuales y los bloques de predicción se emplean entonces para reconstruir bloques de imagen, tal como el bloque de imagen 301. Se debe observar que mientras el método 2000 se describe de la perspectiva del decodificador, puede emplearse un método similar (por ejemplo, en reverso) para codificar el vídeo empleando SVT.

En el bloque 2001, un flujo de bits se obtiene en el decodificador. El flujo de bits puede ser recibido de memoria o de un fuente de retrasmisión. 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 bloque se puede emplear para determinar una unidad de codificado que contiene bloques de predicción y bloques residuales del flujo de bits como se describe en el mecanismo 300. Como tal, información de codificado relacionada con las unidades de codificado se puede analizar sintácticamente del flujo de bits y píxeles de las unidades de codificado se pueden reconstruir en función de la información de codificado como se discute más adelante.

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

En el bloque 2005, se determina el uso de SVT, tipo de SVT y el tamaño de bloque de transformada. Por ejemplo, el decodificador determina primero si SVT se ha usado en la codificación. Esto es porque algunas codificaciones emplean transformadas que son el tamaño del bloque residual. El uso de SVT se puede señalar por un elemento de sintaxis en el flujo de bits. Específicamente, cuando se permite a un bloque residual emplear SVT, un marcador, tal como `svt_flag`,

se analiza sintácticamente del flujo de bits. El bloque residual tiene permitido emplear SVT cuando el bloque residual transformado tiene coeficientes de transformada no cero (por ejemplo, correspondiente a componentes luma o croma). Por ejemplo, un bloque residual puede emplear SVT cuando el bloque residual contiene datos de residual. El marcador de SVT indica si el bloque residual está codificado usando un bloque de transformada de la mismo tamaño al bloque residual (por ejemplo, `svt_flag` se establece a cero) o el bloque residual se codifica con bloque de transformada de un tamaño más pequeño que el bloque residual (por ejemplo, `svt_flag` se establece a uno). Se puede emplear un marcador de bloque codificado (`cbf`) para indicar si un bloque residual incluye coeficientes de transformada no cero de un componente de color, como se usa en HEVC. También, un marcador de bloque codificado raíz (`cbf` de raíz) puede indicar si un bloque residual contiene coeficientes de transformada no cero de cualquier componente de color, como se usa en HEVC. Como ejemplo particular, un bloque residual tiene permitido usar SVT cuando el bloque de imagen se predice usando interpredicción y la anchura de bloque residual o la altura de bloque residual cae en un intervalo predeterminado de $[a1, a2]$, donde $a1=16$ y $a2=64$, $a1=8$ y $a2=64$, o $a1=16$ y $a2=128$. Los valores de $a1$ y $a2$ pueden ser valores fijos predeterminados. Los valores también pueden derivarse de un conjunto de parámetros de secuencia (SPS) o una cabecera de sector en el flujo de bits. Cuando el bloque residual no emplea SVT, el tamaño de bloque de transformada se establece como la anchura y la altura del tamaño de bloque residual. De otro modo, el tamaño de transformada se determina en función del tipo de SVT de transformada.

Una vez el decodificador determina que SVT se ha usado para el bloque residual, el decodificador determina el tipo de bloque de transformada SVT usado y deriva el tamaño de bloque de transformada según el tipo de SVT. Tipos de SVT permitidos para el bloque residual se deciden en función de la anchura y la altura del bloque residual. Una transformada SVT-V como se muestra en la FIGURA 16 se permite si la anchura del bloque residual está en el intervalo $[a1, a2]$ tales valores se definen arriba. Una transformada SVT-H como se muestra en la FIGURA 16 se permite cuando la altura del bloque residual está en el intervalo $[a1, a2]$, tales valores se definen arriba. SVT se puede usar para únicamente los componentes luma en el bloque residual, o SVT puede emplearse para los componentes luma y ambos croma en el bloque residual. Cuando SVT se usa únicamente para el componente luma, la información de componente luma de residual es transformada por SVT y los componentes croma son transformados por una transformada el tamaño del bloque residual. Cuando se permiten ambas SVT-V y SVT-H, un marcador, tal como un `svt_type_flag`, puede codificarse en el flujo de bits. El `svt_type_flag` indica si se usa SVT-V para el bloque residual (por ejemplo, `svt_type_flag` se establece a cero) o SVT-H se usa para el bloque residual (por ejemplo, `svt_type_flag` se establece a uno). Una vez se determina el tipo de transformada SVT, el tamaño de bloque de transformada 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 únicamente se permite SVT-V o únicamente se permite SVT-H, `svt_type_flag` puede no codificarse en el flujo de bits. En tal caso, el decodificador puede inferir el tamaño de bloque de transformada en función del tipo de SVT permitido.

Una vez se determina el tipo de SVT y tamaño, el decodificador procede al bloque 2007. En el bloque 2007, el decodificador determina la posición de la transformada respecto al bloque residual así como el tipo de la transformada (por ejemplo, ya sea DST o DCT). La posición del bloque de transformada se puede determinar según un elemento de sintaxis en el flujo de bits. Por ejemplo, un índice de posición se puede señalar directamente y por tanto analizada sintácticamente del flujo de bits en algunos ejemplos. En otros ejemplos, la posición se puede inferir como se discute con respecto a las FIGURAS 5-7. Específicamente, posiciones candidatas para la transformada se puede determinar según una función CPSS. La función CPSS puede determinar las posiciones candidatas al considerar la anchura del bloque residual, la altura del bloque residual, el tipo de SVT según lo determinado por bloque 2005, el tamaño de etapa de la transformada, y/o la parte del bloque residual que la transformada cubre. El decodificador puede entonces determinar la posición de bloque de transformada de las posiciones candidatas por obtener un p-índice que incluye un código que señala la posición candidata correcta según probabilidad de selección de posición candidata como se discute con respecto a la FIGURA 18 arriba. Una vez se conoce la posición de bloque de transformada, el decodificador puede inferir el tipo de transformada empleado por el bloque de transformada como se discute con respecto a la FIGURA 18A o la FIGURA 18B arriba. Por consiguiente, el codificador puede seleccionar la correspondiente transformada inversa.

En el bloque 2009, el decodificador analiza sintácticamente los coeficientes de transformada del bloque de transformada en función del tamaño de bloque de transformada determinado en bloque 2005. Este proceso puede conseguirse según mecanismos de analizar sintácticamente coeficientes de transformada empleados en HEVC, H.264, y/o codificado de vídeo avanzado (AVC). Los coeficientes de transformada pueden codificarse usando codificado en la longitud (RLE) y/o como conjunto de grupos de coeficientes de transformada (CG). Se debe observar que el bloque 2009 puede realizarse antes que el bloque 2007 en algunos ejemplos.

En el bloque 2011, el bloque residual es reconstruido en función de la posición de transformada, coeficientes de transformada y tipo de transformada como se ha determinado arriba. Específicamente, se aplica cuantización inversa y transformada inversa de tamaño $w_t \times h_t$ a los coeficientes de transformada para recuperar muestras residuales del bloque residual. El tamaño del bloque residual con las muestras residuales es $w_t \times h_t$. La transformada inversa puede ser una DCT inversa o una DST inversa, según la tipo de transformada dependiente de posición determinado en bloque 2007. Las muestras residuales se asignan a una región correspondiente dentro del bloque residual, según la posición de bloque de transformada. Las muestras residuales dentro del bloque residual y fuera del bloque de transformada pueden establecerse a cero. Por ejemplo, cuando se emplea SVT-V, el número de posiciones candidatas es cinco, y el índice de posición indica la quinta posición de bloque de transformada, las muestras residuales

reconstruidas se asignan a la región A en posiciones candidatas de transformada 1700 (e) y la región de tamaño $(w/2) \times h$ a la izquierda de la región A tiene cero muestras residuales.

En el bloque opcional 2013, la información de bloque residual del bloque reconstruido puede voltearse horizontalmente como se discute con respecto a la FIGURA 19. Como se señala anteriormente, esto puede ocurrir cuando el bloque de transformada en el decodificador emplea DST inversa, el bloque vecino del lado derecho ya está reconstruido, y vecino del lado izquierdo aún no está reconstruido. Específicamente, el codificador puede voltear horizontalmente el bloque residual antes de aplicar una transformada DST en el caso mencionado anteriormente a fin de aumentar eficiencia de codificado. Por consiguiente, el bloque opcional 2013 puede emplearse para corregir tal volteo horizontal en el codificador para crear un bloque reconstruido preciso.

En el bloque 2015, el bloque residual reconstruido puede componerse con el bloque de predicción para generar el bloque reconstruido de imagen que contiene muestras como parte de una unidad de codificado. Un proceso de filtrado también puede aplicarse en las muestras reconstruidas, tal como filtro antibloques (deblocking filter) y procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC. El bloque reconstruido de imagen puede entonces combinarse con otros bloques de imagen decodificados de forma similar para generar fotogramas de un archivo multimedia/vídeo. El archivo multimedia reconstruido puede entonces exponerse a un usuario en un monitor u otro dispositivo de exposición.

Se debe observar que una implementación de método 2000 equivalente se pueden emplear para generar las muestras reconstruidas en el bloque residual. Específicamente, las muestras residuales del bloque de transformada se pueden componer directamente con el bloque de predicción en la posición indicada por la información de bloque de transformada de posición, sin recuperar primer el bloque residual.

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

Se hace referencia al siguiente documento y proporcionan además detalles acerca de SVT: C. Zhang, K. Ugur, J. Lainema, A. Hallapuro and M. Gabbouj, "Video Coding Using Spatially Varying Transform", IEEE Trans. Image Process., vol. 21, no.2, Feb. 2011, pp. 127-140.

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

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

En esta realización, SVT tiene restringido ser usada para bloques interpretados. SVT también se puede usar para bloques intrapredichos. En un método más avanzado, SVT puede permitirse para bloques que usan un método de interpretación específico (por ejemplo, compensación de movimiento basada en modelo de traslación) pero no se permite para bloques que usan algún otro método de interpretación (por ejemplo, compensación de movimiento basada en modelo afín). En otro ejemplo, SVT puede permitirse para bloques de predicción que usan modo de fusión o modo AMVP (advanced motion vector prediction) con precisión de diferencia de vector de movimiento 1/4-pel, pero no se permite para bloques de predicción que usan modo de fusión afín, intermodo afín o modo AMVP con precisión de diferencia de vector de movimiento 1-pel o 4-pel. En otro ejemplo, SVT puede permitirse para bloques de predicción que usan modo de fusión con un índice de fusión menor que 2, pero no se permite para bloques de predicción que usan modo de fusión con un índice de fusión no menor que 2. El modo de fusión y el modo AMVP se pueden hacer referencia en H.265/HEVC estándar. A modo de fusión afín e intermodo afín se les puede hacer referencia en códec JEM (Joint Exploration Model) de JVET (Joint Video Exploration Team).

En un ejemplo, un bloque puede referirse a una unidad de codificado, donde la unidad de codificado 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 codificado, el bloque residual puede contener todas las muestras residuales de la unidad de codificado, y el bloque de predicción es del mismo tamaño que el bloque residual. En otro ejemplo, un bloque puede referirse a

una unidad de codificado, una unidad de codificado puede contener dos bloques de predicción y un bloque residual, cada bloque de predicción puede contener una parte de muestras de predicción de la unidad de codificado, y un bloque residual puede contener todas las muestras residuales de la unidad de codificado. En otro ejemplo, un bloque puede referirse a una unidad de codificado, una unidad de codificado puede contener dos bloques de predicción y cuatro bloques residuales. El patrón de partición de bloques residuales en una unidad de codificado puede señalarse en el flujo de bits, tal como el Árbol Cuádruple Residual (RQT) en HEVC.

Un bloque puede contener únicamente componente Y de muestras de imágenes (o píxeles), o puede contener componentes Y, U y V de muestras de imágenes.

Un bloque residual Ro de tamaño $w \times h$ puede ser reconstruido por las siguientes etapas.

Etapla 1. Determinar el tamaño de bloque de transformada del bloque residual Ro.

Etapla 1.1. Determinar el uso de SVT según un elemento de sintaxis. Para un bloque residual que tiene permitido usar SVT, si el bloque residual tiene coeficientes de transformada no cero de componente Y (o tiene coeficientes de transformada no cero de cualquier componente de color), se analiza sintácticamente un marcador (es decir `svt_flag`) del flujo de bits. El marcador indica si el bloque residual está codificado usando un bloque de transformada del mismo tamaño que el bloque residual (por ejemplo, `svt_flag = 0`) o el bloque residual se codifica con un bloque de transformada de un tamaño más pequeño que el bloque residual (por ejemplo, `svt_flag = 1`). Si un bloque tiene coeficientes de transformada no cero de un componente de color puede indicarse por un marcador de bloque codificado (`cbf`) del componente de color, como se usa en HEVC. Si un bloque tiene coeficientes de transformada no cero de cualquier componente de color puede indicarse por un marcador de bloque codificado de raíz (`cbf` de raíz), como se usa en HEVC.

En un ejemplo, un bloque tiene permitido usar SVT si se satisfacen las siguientes condiciones:

- 1) el bloque se predice usando interpredicción;
- 2) ya sea que la anchura de bloque o la altura de 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 de sector.

En otro ejemplo, un bloque tiene permitido usar SVT si se satisfacen las siguientes condiciones.

- 1) el bloque se predice usando modo de fusión con índice de fusión menor que un umbral (por ejemplo, 1 o 2 o 3) o usando modo AMVP con precisión de diferencia de vector de movimiento 1/4-pel;
- 2) una dimensión del bloque cae en un intervalo predeterminado $[a1, a2]$, y la otra dimensión del bloque no es mayor que un umbral $a3$, por ejemplo, $a1=8$, $a2=32$ y $a3=32$. El parámetro $a1$ puede establecerse como el doble del tamaño de transformada mínimo, $a2$ y $a3$ pueden establecerse ambos como el tamaño de transformada máximo. El valor de $a1$, $a2$ y $a3$ pueden ser valores fijos. El valor también se puede derivar de un conjunto de parámetros de secuencia (SPS) o una cabecera de sector.

Si el bloque no usa SVT, el tamaño de bloque de transformada se establece como $w \times h$. De otro modo, se aplica la Etapla 1.2 para decidir el tamaño de transformada.

Etapla 1.2. Determinar el tipo de SVT según un elemento de sintaxis, y derivar el tamaño de bloque de transformada según el tipo de SVT. Tipos de SVT permitidos para el bloque residual se deciden en función de la anchura y la altura del bloque residual. SVT-V se permite si w está en el intervalo $[a1, a2]$ y h no es mayor que $a3$; SVT-H se permite si h está en el intervalo $[a1, a2]$ y w no es mayor que $a3$. SVT puede usarse únicamente para el componente Y, o puede usar para los tres componentes. Cuando SVT se usa únicamente para el componente Y, el componente Y residual se transforma por SVT y los componentes U y V se transforman según el tamaño del bloque residual.

Cuando se permite tanto SVT-V como SVT-H, un marcador (es decir, `svt_type_flag`) se analiza sintácticamente del flujo de bits, que indica si se usa SVT-V (por ejemplo, `svt_type_flag = 0`) o se usa SVT-H (por ejemplo, `svt_type_flag = 1`) para el bloque residual, y el tamaño de bloque de transformada se establece según el tipo de SVT señalizado (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 únicamente se permite SVT-V o únicamente se permite SVT-H, `svt_type_flag` no se analiza sintácticamente del flujo de bits, y el tamaño de bloque de transformada se establece según el tipo de SVT permitido.

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

Etapla 2.1: Determinar la posición de bloque de transformada según un elemento de sintaxis.

Un índice de posición P se analiza sintácticamente del flujo de bits, y el desplazamiento de posición Z de la esquina superior izquierda del bloque de transformada a la esquina superior izquierda del bloque residual se determina como

$$\frac{w - w_{-t}}{s}$$

$Z = s \times P$, donde s es un tamaño de etapa de posición candidata (CPSS). El valor de P está entre 0, 1...,

$$\frac{h - h_{-t}}{s}$$

5 si se usa SVT-V, o el valor de P está entre 0, 1..., $\frac{h - h_{-t}}{s}$ 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 transformada es (Z, 0) para SVT-V o (0, Z) para SVT-H.

10 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 la anchura y la altura del bloque residual, respectivamente, y M1 y M2 son enteros predeterminados en el intervalo de 2 a 8. Se permiten más posiciones candidatas con un valor más grande de M1 o M2. En este ejemplo, M1 y M2 se establecen ambos como 8. Por lo tanto, el valor de P está entre 0 y 4. Las posiciones candidatas se ilustran en la Figura 21 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.

15 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 enteros predefinidos que especifican un tamaño de etapa mínimo. Th1 y Th2 son 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, tamaños de bloque diferentes pueden tener número diferente de posiciones candidatas. Por ejemplo, cuando $w=8$, hay disponibles 2 posiciones candidatas (como se ilustra por la Figura 21(a) y la Figura 21(e)) para elegir; cuando $w=16$, hay disponibles 3 posiciones candidatas (como se ilustra por la Figura 21(a), la Figura 21(c) y la Figura 21(e)) para elegir; cuando $w>16$, hay disponibles 5 posiciones 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. Por consiguiente, se permiten 3 posiciones candidatas.

25 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. Por consiguiente, se permiten 2 posiciones candidatas.

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 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 puede depender además de la relación de aspecto del bloque residual.

35 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, conjunto de parámetros de secuencia). M1 y M2 pueden compartir el mismo valor analizado sintácticamente de un elemento de sintaxis, y Th1 y Th2 puede compartir el mismo valor analizado sintácticamente de otro elemento de sintaxis.

40 El índice de posición P puede binarizarse en uno o más bins usando código unario truncado. Por ejemplo, si el valor P está en el intervalo de 0 a 4, el valor 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, el valor P 0 y 1 se binarizan como 0 y 1, respectivamente.

45 El índice de posición P puede binarizarse en uno o más bins usando la posición más probable y varias posiciones restantes. Cuando los vecinos izquierdo y superior están disponibles, la posición más probable puede establecerse 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, el valor 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, el valor P 2, 0 y 1 se binarizan como 1, 01 y 00, respectivamente.

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

55 Tomando el caso de que se permiten 3 posiciones candidatas para un ejemplo, 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 ambas SVT-V y SVT-H, como se muestra en la Figura 22.

60 Una transformada bidimensional puede ser separable en transformada horizontal y transformada vertical unidimensionales. Una transformada 2D hacia delante que convierte el residual en coeficientes de transformada puede realizarse aplicando primero transformada horizontal en un bloque residual para generar un bloque TA y entonces

aplicando transformada vertical en el bloque TA para generar un bloque de coeficiente de transformada, como se implementa en códec JEM. Por consiguiente, una transformada 2D inversa que convierte coeficientes de transformada de nuevo en residual puede realizarse aplicando primero transformada vertical inversa en un bloque de coeficiente de transformada para generar un bloque TB y entonces aplicando transformada horizontal inversa en el bloque TB para generar un bloque residual, como se implementa en códec JEM.

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

Tabla I. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en un ejemplo, las transformadas horizontal y vertical para SVT-V posición 0 son DCT-4 y DST-4; las transformadas horizontal y vertical para SVT-V posición 1 son DST-1 y DST-4; las transformadas horizontal y vertical para SVT-V posición 2 son DST-4 y DST-4; las transformadas horizontal y vertical para SVT-H posición 0 son DST-4 y DCT-4; las transformadas horizontal y vertical para SVT-H posición 1 son DST-4 y DST-1; las transformadas horizontal y vertical para SVT-H posición 2 son DST-4 y DST-4, como se enumera en la Tabla I'. En este ejemplo, la transformada vertical para SVT-V y la transformada horizontal para SVT-H se establecen como DST-4, y las otras transformadas se basan en posición de SVT.

Tabla I'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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 transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla II. En este ejemplo, la transformada vertical para SVT-V y la transformada horizontal para SVT-H se establecen como DCT-2, y las otras transformadas se basa en posición SVT.

Tabla II. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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

Tipo SVT, posición	transformada horizontal	transformada vertical
SVT-H, posición 1	DCT-2	DST-1
SVT-H, posición 2	DCT-2	DST-7

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla II'. En este ejemplo, la transformada vertical para SVT-V y la transformada horizontal para SVT-H se establecen como DCT-2, y las otras transformadas se basa en posición SVT.

Tabla II'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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

En otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla III. En este ejemplo, la transformada horizontal y la transformada vertical se deciden únicamente por posición de SVT.

Tabla III. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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 manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla III'. En este ejemplo, la transformada horizontal y la transformada vertical se deciden únicamente por posición de SVT.

Tabla III'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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
SVT-H, posición 2	DST-4	DST-4

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

Tabla IV. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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

- 5 En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla IV'.

Tabla IV'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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 transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla V'.

Tabla V. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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

En una manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla V'.

Tabla V'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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
SVT-H, posición 1	DST-4	DCT-2

Tipo SVT, posición	transformada horizontal	transformada vertical
SVT-H, posición 2	DST-4	DST-4

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

5 Tabla VI. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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 manera de implementación alternativa, DST-7 puede sustituirse con DST-4, y DCT-8 puede sustituirse con DCT-4 por toda la presente divulgación. Por consiguiente, en otro ejemplo, la transformada horizontal y la transformada vertical para diferentes tipos y posiciones de SVT se enumeran en la Tabla VI'.

10

Tabla VI'. Un ejemplo de transformadas horizontal y vertical 1-D para diferentes tipos y posiciones de SVT.

Tipo SVT, posición	transformada horizontal	transformada 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 múltiples transformadas dependientes de posición pueden aplicarse a bloque de transformada luma únicamente, y los correspondientes bloques de transformada croma siempre usan DCT-2 inversa en el proceso de transformada inversa.

20 Etapa 3. Analizar sintácticamente coeficientes de transformada del bloque de transformada en función del tamaño de bloque de transformada.

Este es un proceso usado comúnmente en decodificación de vídeo, tal como el análisis sintáctico de coeficientes de transformada en HEVC o H.264/AVC. Los coeficientes de transformada pueden codificarse usando codificado en la longitud o codificado más sofisticado como conjunto de grupos de coeficientes de transformada (CG).

25

La Etapa 3 puede realizarse antes que la Etapa 2.

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

30

Cuantización inversa y transformada inversa de tamaño $w_t \times h_t$ se aplican a los coeficientes de transformada para recuperar muestras residuales. El tamaño de las muestras residuales es $w_t \times h_t$, que es el mismo que el tamaño de bloque de transformada. La transformada inversa es una transformada separable 2-D. El bloque de coeficiente de transformada descuantizado se transforma primero por una transformada vertical inversa para generar un bloque TC y entonces el bloque TC se transforma por una transformada horizontal inversa, en la que la transformada horizontal inversa y la transformada vertical inversa se deciden en la etapa 2.2 en función de posición de bloque de transformada o en función de ambos posición de bloque de transformada y tipo de SVT del bloque de transformada.

35

Las muestras residuales se asignan a una región correspondiente dentro del bloque residual R_0 , según la posición de bloque de transformada, y las muestras restantes dentro del bloque residual se establecen para ser cero. Por ejemplo,

40

si se usa SVT-V y el número de posición candidata es 5 y el índice de posición es 4, las muestras residuales reconstruidas se asignan a 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 Tras realizar de la Etapa 1 a la Etapa 4, el bloque residual reconstruido puede componerse con un bloque de predicción para generar las muestras reconstruidas en una unidad de codificado. Más tarde puede aplicarse un proceso de filtrado en las muestras reconstruidas, tal como filtro antibloques (deblocking filter) y procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC.

10 A diferencia de las soluciones existentes, la presente solución usa adaptativamente múltiples tipos de transformada para bloque de transformada en función del tipo de SVT e información de posición.

La presente divulgación introduce un esquema de SVT mejorado. Se usan dos tipos de bloques de SVT para codificado de residual, denotados como SVT-H y SVT-V, como se muestra en la Figura 23 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención. Hay (1) SVT-V: $w_t = w/2$ 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 transformada en SVT-V y SVT-H se agranda a la mitad del bloque residual, que puede cubrir más residual en el bloque residual. Obsérvese que SVT-II o SVT-H se pueden usar para sustituir a SVT-V o SVT-H.

Posiciones candidatas se deciden por un Tamaño de Etapa de la posición Candidata (CPSS). Por consiguiente, 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 realizaciones, lo que mitiga la sobrecarga de la información de posición así como la complejidad del codificador para decidir la mejor posición de bloque de transformada.

El CPSS denota la distancia entre dos posiciones candidatas adyacentes (por ejemplo, 1 en [1]), y se calcula en función de la anchura o altura del bloque residual. Más específicamente, el tamaño de etapa de posición candidata para SVT-V se determina como $h_b = \max(w/M1, Th1)$, donde M1 y Th1 son enteros predefinidos, y $\max(a,b)$ es la operación de seleccionar el valor máximo entre a y b. De manera similar, el tamaño de etapa de posición candidatas para SVT-H se determina como $w_b = \max(w/M2, Th2)$, donde M2 y Th2 son enteros predefinidos. M1 y M2 son enteros entre 2 y 8, por ejemplo, $M1=M2=8$, o $M1=M2=4$, o $M1=M2=2$. Th1 y Th2 son 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, conjunto de parámetros de secuencia).

El codificado de información de SVT (por ejemplo, el uso de SVT para un bloque residual, el tipo de SVT y la posición del bloque de transformada) es diferente de la de [1], y además se mejora al analizar los coeficientes de transformada del bloque residual.

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

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

50 Configuración 1

Esta configuración describe el proceso de decodificar un bloque residual. Se decodifica un flujo de bits que contiene al menos una imagen de datos de vídeo. Una imagen se parte en una pluralidad de regiones de imagen rectangulares y cada región corresponde a una unidad de Árbol de Codificado (CTU). Una CTU se particiona en una pluralidad de bloques, tales como las unidades de codificado en Codificado de Vídeo de Alta Eficiencia (HEVC), según información de partición de bloque contenida en el flujo de bits. Información de codificado de los bloques se analiza sintácticamente del flujo de bits y píxeles de los bloques se reconstruyen en función de la información de codificado.

60 En esta configuración, SVT se restringe para ser usada para bloques interpredichos. SVT también se puede usar para bloques intrapredichos. En 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 se permite para bloques que usan algún otro método de interpredicción (por ejemplo, compensación de movimiento basada en modelo afín).

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

Un bloque puede contener únicamente componente Y de muestras de imágenes (o píxeles), o puede contener componentes Y, U y V de muestras de imágenes.

Un bloque residual Ro de tamaño w×h puede ser reconstruido por las siguientes etapas.

Etapa 1. Determinar el tamaño de bloque de transformada del bloque residual Ro.

Etapa 1.1. Determinar el uso de SVT según un elemento de sintaxis. Para un bloque residual que tiene permitido usar SVT, si el bloque residual tiene coeficientes de transformada no cero de componente Y (o tiene coeficientes de transformada no cero de cualquier componente de color), un marcador (es decir svt_flag) se analiza sintácticamente del flujo de bits, que indica si el bloque residual está codificado usando un bloque de transformada del mismo tamaño que el bloque residual (por ejemplo, svt_flag = 0) o el bloque residual está codificado con un bloque de transformada de un tamaño más pequeño que el bloque residual (por ejemplo, svt_flag = 1). Si un bloque tiene coeficientes de transformada no cero de un componente de color puede indicarse por un marcador de bloque codificado (cbf) del componente de color, como se usa en HEVC. Si un bloque tiene coeficientes de transformada no cero de cualquier componente de color puede indicarse por un marcador de bloque codificado de raíz (cbf de raíz), como se usa en HEVC.

En un ejemplo, un bloque tiene permitido usar SVT si se satisfacen las siguientes condiciones.

- 1) el bloque se predice usando interpredicción;
- 2) ya sea que la anchura de bloque o la altura de 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 de sector.

Si el bloque no usa SVT, el tamaño de bloque de transformada se establece como w×h. De otro modo, se aplica la Etapa 1.2 para decidir el tamaño de transformada.

Etapa 1.2. Determinar el tipo de SVT según un elemento de sintaxis, y derivar el tamaño de bloque de transformada según el tipo de SVT. Tipos de SVT permitidos para el bloque residual se deciden en función de la anchura y la altura del bloque residual. SVT-V se permite si w está en el intervalo [a1, a2]; SVT-H se permite si h está en el intervalo [a1, a2]. SVT puede usarse únicamente para el componente Y, o puede usar para los tres componentes. Cuando SVT se usa únicamente para el componente Y, el componente Y residual se transforma por SVT y los componentes U y V se transforman según el tamaño del bloque residual.

Cuando se permite tanto SVT-V como SVT-H, un marcador (es decir, svt_type_flag) se analiza sintácticamente del flujo de bits, que indica si se usa SVT-V (por ejemplo, svt_type_flag = 0) o se usa SVT-H (por ejemplo, svt_type_flag = 1) para el bloque residual, y el tamaño de bloque de transformada 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 únicamente se permite SVT-V o únicamente se permite SVT-H, svt_type_flag no se analiza sintácticamente del flujo de bits, y el tamaño de bloque de transformada se establece según el tipo de SVT permitido.

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

Un índice de posición P se analiza sintácticamente del flujo de bits, y el desplazamiento de posición Z de la esquina superior izquierda del bloque de transformada a la esquina superior izquierda del bloque residual se determina como

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

$$0, 1, \dots, \frac{h - h_t}{s} \text{ si se usa SVT-H. Más específicamente, si } (0, 0) \text{ representa la coordenada de la esquina superior izquierda del bloque residual, la coordenada de la esquina superior izquierda del bloque de transformada es } (Z, 0) \text{ para SVT-V o } (0, Z) \text{ para SVT-H.}$$

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 la anchura y la altura del bloque residual, respectivamente, y $M1$ y $M2$ son enteros predeterminados en el intervalo de 2 a 8. Se permiten más posiciones candidatas con un valor más grande de $M1$ o $M2$. En este ejemplo, $M1$ y $M2$ se establecen ambos como 8. Por lo tanto, el valor de P está entre 0 y 4. Las posiciones candidatas se ilustran en la Figura 24 como ejemplo que no comprende todos los rasgos necesarios para implementar la presente invención.

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 enteros predefinidos que especifican un tamaño de etapa mínimo. $Th1$ y $Th2$ son 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, tamaños de bloque diferentes pueden tener número diferente de posiciones candidatas. Por ejemplo, cuando $w=8$, hay disponibles 2 posiciones candidatas (como se ilustra por la Figura 24(A) y la Figura 24(E)) para elegir; cuando $w=16$, hay disponibles 3 posiciones candidatas (como se ilustra por la Figura 24(A), la Figura 24(C) y la Figura 24(E)) para elegir; cuando $w>16$, hay disponibles 5 posiciones 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. Por consiguiente, 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. Por consiguiente, 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 puede depender 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, conjunto de parámetros de secuencia). $M1$ y $M2$ pueden compartir el mismo valor analizado sintácticamente de un elemento de sintaxis, y $Th1$ y $Th2$ puede compartir el mismo valor analizado sintácticamente de otro elemento de sintaxis.

El índice de posición P puede binarizarse en uno o más bins usando código unario truncado. Por ejemplo, si el valor P está en el intervalo de 0 a 4, el valor 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, el valor P 0 y 1 se binarizan como 0 y 1, respectivamente.

Etapa 3. Analizar sintácticamente coeficientes de transformada del bloque de transformada en función del tamaño de bloque de transformada.

Este es un proceso usado comúnmente en decodificación de vídeo, tal como el análisis sintáctico de coeficientes de transformada en HEVC o H.264/codificado de vídeo avanzado (AVC). Los coeficientes de transformada pueden codificarse usando codificado en la longitud o codificado más sofisticado como conjunto de grupos de transformadas.

La Etapa 3 puede realizarse antes que la Etapa 2.

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

Cuantización inversa y transformada inversa de tamaño $w_t \times h_t$ se aplican a los coeficientes de transformada para recuperar muestras residuales. El tamaño de las muestras residuales es $w_t \times h_t$, que es el mismo que el tamaño de bloque de transformada. La transformada inversa puede ser un Transformada Discreta de Coseno (DCT) inversa o una Transformada Discreta de Seno (DST) inversa.

Las muestras residuales se asignan a una región correspondiente dentro del bloque residual R_0 , según la posición de bloque de transformada, y las muestras restantes dentro del bloque residual se establecen para ser cero. Por ejemplo, si se usa SVT-V y el número de posición candidata es 5 y el índice de posición es 4, las muestras residuales reconstruidas se asignan a 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.

Tras realizar de la Etapa 1 a la Etapa 4, el bloque residual reconstruido puede componerse con un bloque de predicción para generar las muestras reconstruidas en una unidad de codificado. Más tarde puede aplicarse un proceso de filtrado en las muestras reconstruidas, tal como filtro antibloques (deblocking filter) y procesamiento de desplazamiento adaptativo de muestras (SAO) en HEVC.

Configuración 2

En otra configuración, la información de transformada de posición se deriva de los coeficientes de transformada del bloque residual. Un bloque residual Ro de tamaño $w \times h$ puede ser reconstruido por las siguientes etapas.

Etapas 1. Determinar el tamaño de bloque de transformada del bloque residual Ro.

Se puede aplicar el mismo proceso descrito en la etapa 1 de la configuración 1.

Etapas 2. Analizar sintácticamente coeficientes de transformada del bloque de transformada en función del tamaño de bloque de transformada.

Se puede aplicar el mismo proceso descrito en la etapa 3 de la configuración 1.

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

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

El índice de posición P (que indica la posición de bloque de transformada, como se describe en la configuración 1) se deriva por la suma de los coeficientes de transformada St y el número de posiciones candidatas Nt asociadas con el bloque de transformada, según la siguiente fórmula:

$$P = St \% Nt,$$

donde % denota una operación de modulo que encuentra el resto tras división de St por Nt.

Más específicamente, en un ejemplo, los parámetros M1 y M2 se establecen como 4, y hay 3 posiciones candidatas para SVT-H o SVT-V (es decir, Nt 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 M1 y M2 se establecen como 2, y hay 2 posiciones candidatas para SVT-H o SVT-V (es decir, Nt 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 ser parcialmente decidido por la suma de coeficientes de transformada. Los parámetros M1 y M2 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 otro modo, el valor P está entre 1, 2 y 3 (pero no decidido aún). Entonces, un elemento de sintaxis que tiene tres valores posibles se analiza sintácticamente para determinar si P es igual a 1 o 2 o 3 en función del valor de la elemento de sintaxis (por ejemplo, P se establece como 1, 2 o 3 si el valor de elemento de sintaxis es 0, 1 o 2).

Puede haber alguna comprobación de condición para determinar si el índice de posición información está oculto en la suma de coeficientes. Por ejemplo, cuando el número de coeficientes de transformada es menor que un umbral (por ejemplo, 4 o 6) o la distancia entre el primero y los últimos coeficientes son menor que un umbral (por ejemplo, 5 o 6), el índice de posición no depende de la suma de coeficientes y se señala explícitamente, y el proceso de analizar sintácticamente en la configuración 1 se aplica para determinar el índice de posición.

Etapas 4. Reconstruir el bloque residual Ro en función de los coeficientes de transformada y la posición de bloque de transformada.

Se puede aplicar el mismo proceso descrito en la etapa 4 de la Realización 1.

Configuración 3

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

Un bloque residual Ro de tamaño $w \times h$ puede ser reconstruido por las siguientes etapas.

Etapas 1. Determinar el tamaño de bloque de transformada del bloque residual Ro, por la etapa 1.1 y la etapa 1.2.

Etapas 1.1. Determinar el uso de SVT. Esta es igual que la Etapa 1.1 en la configuración 1.

Etapas 1.2. Analizar sintácticamente un primer grupo de coeficientes de transformada y determinar el tipo de SVT en función de la suma del primer grupo de coeficientes de transformada.

El primer grupo de coeficientes de transformada puede estar en una región de tamaño $w_{t1} \times h_{t1}$ en la parte superior izquierda del bloque de transformada (correspondiente a menores coeficientes de transformada de frecuencia), como se ilustra por la región sólida 1 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 $h_{t1} = \max(h/8, 4)$.

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

Etapas 2. Analizar sintácticamente los coeficientes de transformada restantes en el bloque de transformada en función del tamaño de bloque de transformada.

El orden de escaneo de coeficientes se deriva en función de la anchura y la altura del bloque de transformada. Cuando se analizan sintácticamente los coeficientes de transformada, los coeficientes en posiciones cubre por el primer grupo de coeficientes de transformada se omiten y se asignan para ser el primer grupo de coeficientes de transformada.

Etapas 3. Derivar la posición de bloque de transformada en función de la suma de un segundo grupo de coeficientes de transformada.

Un segundo grupo de coeficientes de transformada se usa para calcular una suma. En función de la suma, el índice de posición se deriva, en el que se puede usar el método descrito en la configuración 2. El segundo grupo de coeficientes de transformada puede estar en una región de tamaño $w_{t2} \times h_{t2}$ en la parte superior izquierda del bloque de transformada aparte del primer grupo de coeficientes de transformada, como se ilustra por la región de puntos 2 en la Figura 25. En un ejemplo, $w_{t2} = w_t$ y $h_{t2} = h_t$, es decir, todos los coeficientes aparte de los coeficientes de primer grupo pertenecen al segundo grupo. En otro ejemplo, $w_{t2} = \max(w_t/2, w_{t1})$ y $h_{t2} = \max(h_t/2, h_{t1})$.

Etapas 4. Reconstruir el bloque residual R_o en función de los coeficientes de transformada y la posición de bloque de transformada. Se puede usar el procesamiento en la etapa 4 de la configuración 1.

Configuración 4

En otra configuración, el número de posiciones candidatas para un bloque residual usando SVT se deriva en función de la distribución de coeficiente de transformada.

Un bloque residual R_o de tamaño $w \times h$ puede ser reconstruido por las siguientes etapas.

Etapas 1. Determinar el tamaño de bloque de transformada del bloque residual R_o .

Se puede aplicar el mismo proceso descrito en la etapa 1 de la Realización 1.

Etapas 2. Analizar sintácticamente coeficientes de transformada del bloque de transformada en función del tamaño de bloque de transformada.

Se puede aplicar el mismo proceso descrito en la etapa 3 de la configuración 1.

Etapas 3. Derivar la posición de bloque de transformada en función de la distribución de los coeficientes de transformada.

En un ejemplo, el número de coeficientes de transformada no cero N_{nz} se usan para decidir el número de posiciones candidatas. Si N_{nz} es menor que un umbral (por ejemplo, 4 o 5 o 8), el tamaño de etapa de posición candidata se establece como $w/4$ para SVT-V y $h/4$ para SVT-H, y así el número de posiciones candidatas es 3; de otro modo, el tamaño de etapa de posición candidata se establece como $w/8$ para SVT-V y $h/8$ o SVT-H, y así el número de posiciones candidatas es 5.

En otro ejemplo, si los últimos coeficientes de transformada no cero están dentro del área superior izquierda 4×4 en el bloque de transformada, el tamaño de etapa de posición candidata se establece como $w/4$ para SVT-V y $h/4$ para SVT-H, y así el número de posiciones candidatas es 3; de otro modo, el tamaño de etapa de posición candidata se establece como $w/8$ para SVT-V y $h/8$ o SVT-H, y así el número de posiciones candidatas es 5.

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

Etapas 4. Reconstruir el bloque residual R_o en función de los coeficientes de transformada y la posición de bloque de transformada. Se puede usar el procesamiento en la etapa 4 de la configuración 1.

La presente divulgación proporciona los siguientes beneficios y ventajas respecto a soluciones conocidas.

Las posiciones candidatas para bloques de SVT se deciden por un tamaño de etapa relacionado con la anchura y la altura del bloque residual y se limitan para no ser más de 5, lo que reduce la información de bloque de transformada en comparación con [1] y logra un mejor equilibrio entre sobrecarga de señalización y eficiencia de codificado de residual.

La información de SVT se codifica diferente a [1], es decir, se organiza como 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 transformada del bloque residual.

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

Un experto en la técnica puede entender claramente que, a efectos de una descripción conveniente y breve, 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 de acuerdo con un requisito, 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, aparato y unidad anteriores se puede hacer referencia a un proceso correspondiente en las realizaciones del método anteriores, y estos detalles no se vuelven a describir en esta memoria.

En las varias realizaciones proporcionadas en la presente solicitud, se debe entender que el sistema, aparato y método descritos se pueden implementar de otras maneras. Por ejemplo, la realización del aparato descrita es meramente ejemplar. Por ejemplo, la división de módulo o unidad es meramente una división de función lógica y en la implementación real puede ser otra división. Por ejemplo, una pluralidad de unidades o componentes se pueden combinar o integrar en otro sistema, o algunas funciones se pueden ignorar o se pueden no realizar. Además, los acoplamientos mutuos o los acoplamientos directos o las conexiones de comunicación mostrados o discutidos se pueden implementar utilizando algunas interfaces. Los acoplamientos indirectos o las conexiones de comunicación entre los aparatos o unidades se pueden implementar de forma electrónica, mecánica o de otro tipo.

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

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 se puede disponer físicamente de manera individual, o dos o más unidades se integran en una unidad. La unidad integrada se puede implementar en forma de hardware, o se puede implementar en forma de una unidad funcional de software.

Cuando la unidad integrada se implementa en forma de una unidad funcional de software y se vende o utiliza como producto independiente, la unidad integrada se puede almacenar en un soporte de almacenamiento legible por ordenador. En función de dicha interpretación, las soluciones técnicas de la presente invención, o la parte que contribuye a la técnica anterior, o todas o algunas de las soluciones técnicas se pueden implementar en forma de un producto de software. El producto de software informático se almacena en un soporte de almacenamiento e incluye varias instrucciones para instruir a un dispositivo informático (que puede ser un ordenador personal, un servidor o un dispositivo de red) o un procesador (procesador) para realizar todas o algunas de las etapas de los métodos descritos en las realizaciones de la presente invención. El soporte de almacenamiento anterior incluye: cualquier soporte que pueda almacenar código de programa, tal como una unidad flash USB, un disco duro extraíble, una memoria de solo lectura (ROM, Read-Only Memory), una memoria de acceso aleatorio (RAM, Random Access Memory), un disco magnético o un disco óptico.

Las descripciones anteriores son maneras de implementación meramente específicas de la presente invención, pero no pretenden limitar la protección de la presente invención. Cualquier variación o sustitución ideada fácilmente por un experto en la técnica dentro de la solución técnica de las reivindicaciones, estará dentro de la protección de la presente invención. Por lo tanto, la protección de la presente invención estará sujeto a la de protección de las reivindicaciones.

REIVINDICACIONES

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

- 5 analizar sintácticamente (1201) un flujo de bits recibido para obtener información de predicción de una unidad de codificado, CU;
cuando la CU tiene únicamente una unidad de transformada residual, TU, y un tamaño de la TU residual es menor que un tamaño de la CU, obtener (1202, 1202') un objetivo de modo de transformada de la TU residual;
10 en donde el objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual, y un conjunto de tipos de transformada de la TU residual; en donde las TU restantes de la CU se infieren para no tener residual;
obtener (1202') un tipo de transformada de la TU residual del conjunto de tipos de transformada;
analizar sintácticamente (1203) el flujo de bits recibido para obtener coeficientes de transformada de la TU residual;
15 aplicar (1204) una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados;
aplicar (1205), en función del tipo de transformada de la TU residual o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual, una transformada inversa a los coeficientes descuantizados para obtener un bloque residual de la TU residual;
20 obtener (1206) un bloque de predicción de la CU en función de la información de predicción;
obtener (1207) un bloque de vídeo en función del bloque residual y el bloque de predicción; y
sacar (1208) una secuencia de vídeo, la secuencia de vídeo que incluye un fotograma de vídeo que incluye el bloque de vídeo;
25 en donde el tamaño de la CU se denota por $W \times H$; en donde el objetivo de modo de transformada de la TU residual se selecciona de un conjunto que comprende los siguientes modos:
- modo de transformada HQ0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-7; en donde el tamaño de la TU superior se denota por $W \times (H/4)$ y el tamaño de la TU inferior se denota por $W \times (3H/4)$;
- 30 modo de transformada HQ1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8; en donde el tamaño de la TU inferior se denota por $W \times (H/4)$ y el tamaño de la TU superior se denota por $W \times (3H/4)$;
- 40 modo de transformada VQ0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7, y una transformada vertical de la TU residual es DCT-8 o DCT-2; en donde el tamaño de la TU izquierda se denota por $(W/4) \times H$ y el tamaño de la TU derecha se denota por $(3W/4) \times H$;
- 50 modo de transformada VQ1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8, y una transformada vertical de la TU residual es DCT-8 o DCT-2; en donde el tamaño de la TU derecha se denota por $(W/4) \times H$ y el tamaño de la TU izquierda se denota por $(3W/4) \times H$;
- 60 modo de transformada HH0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en
- 65

donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-7;

modo de transformada HH1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8;

modo de transformada VH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-8 o DCT-2; y

modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DCT-8 o DCT-2.

2. El método según la reivindicación 1, en donde la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende:

seleccionar un tipo de transformada de la TU residual del conjunto de tipos de transformada.

3. El método según una cualquiera de las reivindicaciones anteriores, el método comprende además: analizar sintácticamente información de indicación del flujo de bits, en donde la información de indicación indica un tipo de transformada de la TU residual;

la etapa de obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, comprende:

determinar un tipo de transformada de la TU residual del conjunto de tipos de transformada en función de la información de indicación analizada sintácticamente del flujo de bits.

4. El método según una cualquiera de las reivindicaciones anteriores, en donde obtener el objetivo de modo de transformada de la TU residual comprende:

analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada de la TU residual; y
obtener el objetivo de modo de transformada de la TU residual en función del índice de modo.

5. Un decodificador de vídeo (30), que comprende:

una unidad de decodificación por entropía (304), se configura para analizar sintácticamente un flujo de bits recibido para obtener información de predicción de una unidad de codificado, (CU; obtener un objetivo de modo de transformada de una unidad de transformada residual, TU, 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 objetivo de modo de transformada especifica un modo de particionamiento de TU de la CU, la posición de la TU residual, y un conjunto de tipos de transformada de la TU residual; y analizar sintácticamente el flujo de bits recibido para obtener coeficientes de transformada de la TU residual; en donde las TU restantes de la CU se infieren para no tener residual;

en donde la unidad de decodificación por entropía (304) se configura además para obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada;

una unidad de procesamiento de cuantización inversa (310), se configura para aplicar una cuantización inversa a los coeficientes de transformada de la TU residual para obtener coeficientes descuantizados;

una unidad de procesamiento de transformada inversa (312), se configura para aplicar, en función del tipo de transformada de la TU residual o en función del modo de particionamiento de TU de la CU, la posición de la TU residual y el tipo de transformada de la TU residual, una transformada inversa a los coeficientes descuantizados para obtener un bloque residual de la TU residual;

una unidad de procesamiento de predicción (360), se configura para obtener un bloque de predicción de la CU en función de la información de predicción;

una unidad de reconstrucción (314), se configura para obtener un bloque de vídeo en función del bloque residual y el bloque de predicción; y

5 una salida (332), se configura para tener como salida una secuencia de vídeo, la secuencia de vídeo incluye un fotograma de vídeo que incluye el bloque de vídeo;

en donde el tamaño de la CU se denota por $W \times H$; en donde el objetivo de modo de transformada de la TU residual se selecciona de un conjunto que comprende los siguientes modos:

10 modo de transformada HQ0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-7; en donde el tamaño de la TU superior se denota por $W \times (H/4)$ y el tamaño de la TU inferior se denota por $W \times (3H/4)$;

20 modo de transformada HQ1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8; en donde el tamaño de la TU inferior se denota por $W \times (H/4)$ y el tamaño de la TU superior se denota por $W \times (3H/4)$;

30 modo de transformada VQ0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7, y una transformada vertical de la TU residual es DCT-8 o DCT-2; en donde el tamaño de la TU izquierda se denota por $(W/4) \times H$ y el tamaño de la TU derecha se denota por $(3W/4) \times H$;

35 modo de transformada VQ1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-8, y una transformada vertical de la TU residual es DCT-8 o DCT-2; en donde el tamaño de la TU derecha se denota por $(W/4) \times H$ y el tamaño de la TU izquierda se denota por $(3W/4) \times H$;

40 modo de transformada HH0, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-8; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-7;

50 modo de transformada HH1, que especifica que el modo de particionamiento de TU de la CU es para particionar 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 transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DST-7 o DCT-2, y una transformada vertical de la TU residual es DCT-7; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DCT-8 o DCT-2, y una transformada vertical de la TU residual es DST-8;

60 modo de transformada VH0, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es izquierda, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU

residual es DCT-8 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que una transformada horizontal de la TU residual es DST-7 y una transformada vertical de la TU residual es DCT-8 o DCT-2; y

5 modo de transformada VH1, que especifica que el modo de particionamiento de TU de la CU es para particionar la CU en 2 TU, en donde un tamaño de una TU izquierda es igual a un tamaño de una TU derecha, la posición de la TU residual es derecha, y el tipo de transformada se selecciona de un conjunto de tipos de transformada; el conjunto de tipos de transformada incluye dos tipos de transformada, en donde un primer tipo de transformada es que una transformada horizontal de la TU residual es DCT-7 y una transformada vertical de la TU residual es DST-7 o DCT-2; un segundo tipo de transformada es que
10 una transformada horizontal de la TU residual es DST-8 y una transformada vertical de la TU residual es DCT-8 o DCT-2.

6. El decodificador de vídeo (30) según la reivindicación 5, en donde para obtener un tipo de transformada de la TU residual del conjunto de tipos de transformada, la unidad de decodificación por entropía (304) se configura
15 específicamente para seleccionar un tipo de transformada de la TU residual del conjunto de tipos de transformada.

7. El decodificador de vídeo (30) según una cualquiera de las reivindicaciones anteriores, en donde la unidad de decodificación por entropía (304) se configura además para analizar sintácticamente información de indicación del flujo de bits, en donde la información de indicación indica un tipo de transformada de la TU residual; y
20 determinar un tipo de transformada de la TU residual del conjunto de tipos de transformada en función de la información de indicación analizada sintácticamente del flujo de bits.

8. El decodificador de vídeo (30) según una cualquiera de las reivindicaciones anteriores, en donde la unidad de decodificación por entropía (304) se configura además para analizar sintácticamente el flujo de bits para obtener un índice de modo para indicar el objetivo de modo de transformada de la TU residual; y obtener el objetivo de modo de transformada de la TU residual en función del índice de modo.
25

9. Un producto de programa informático que comprende un código de programa para realizar el método según una cualquiera de las reivindicaciones 1 a 4.

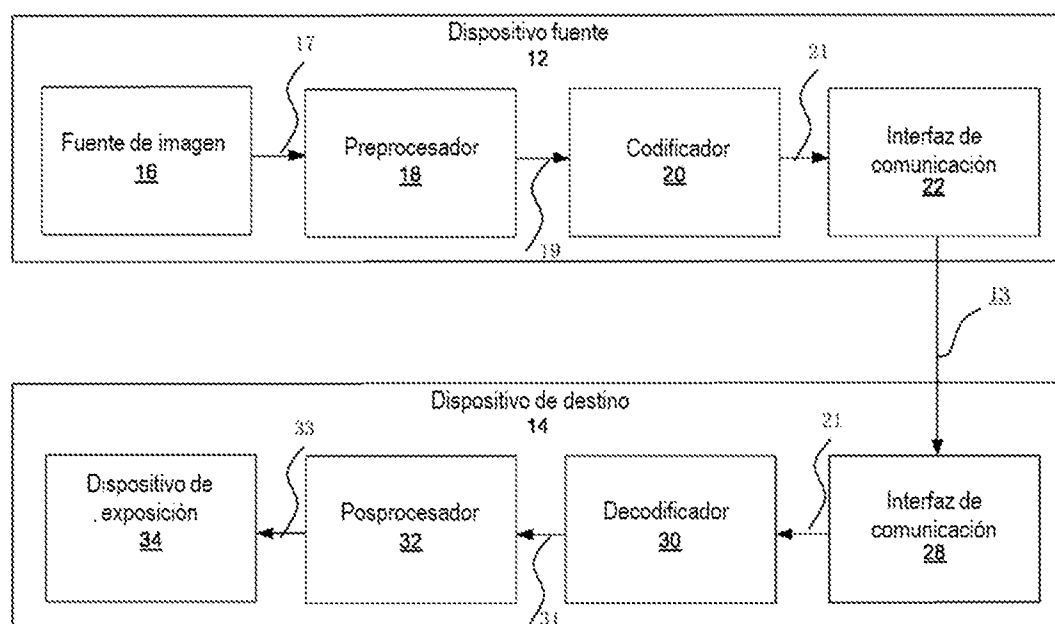


Fig. 1A

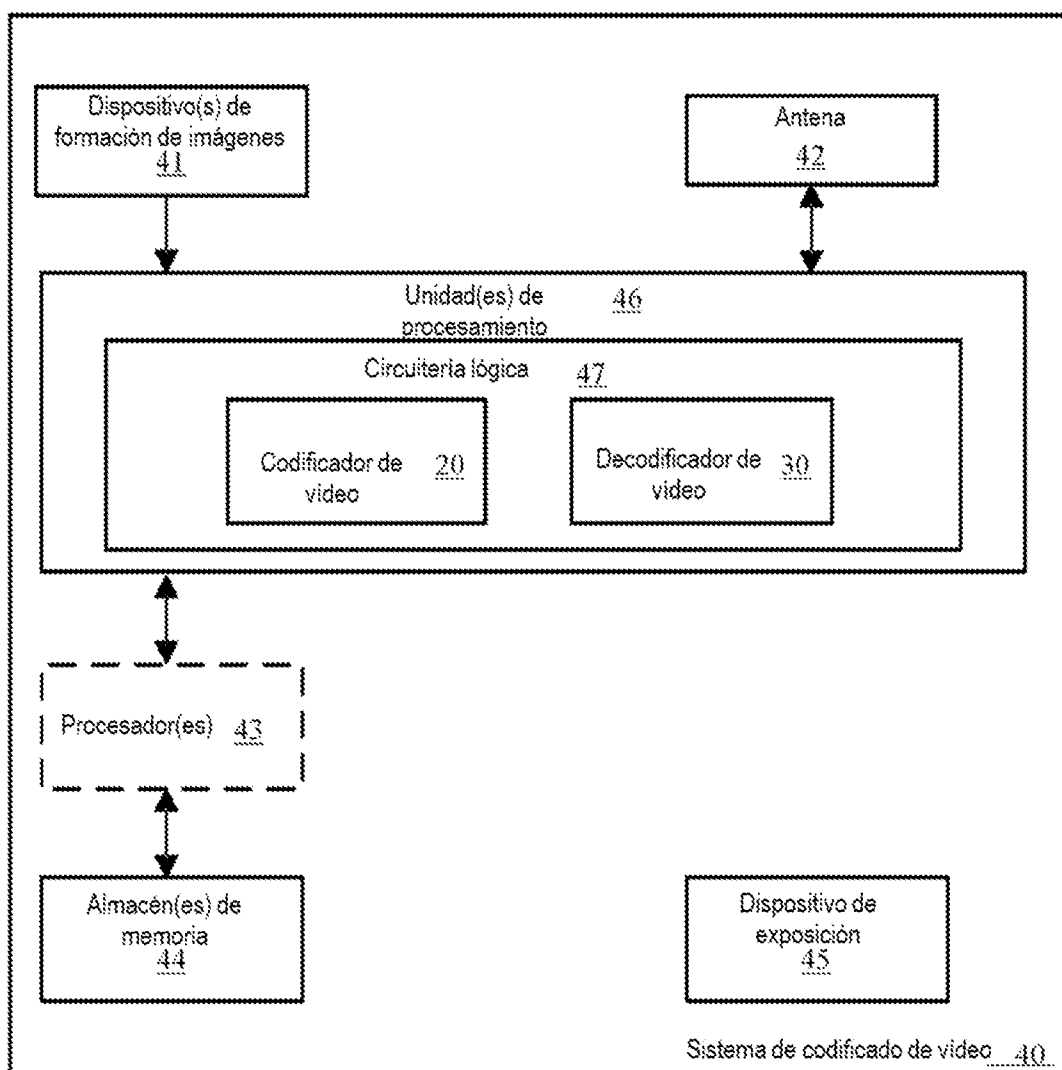


Fig. 1B

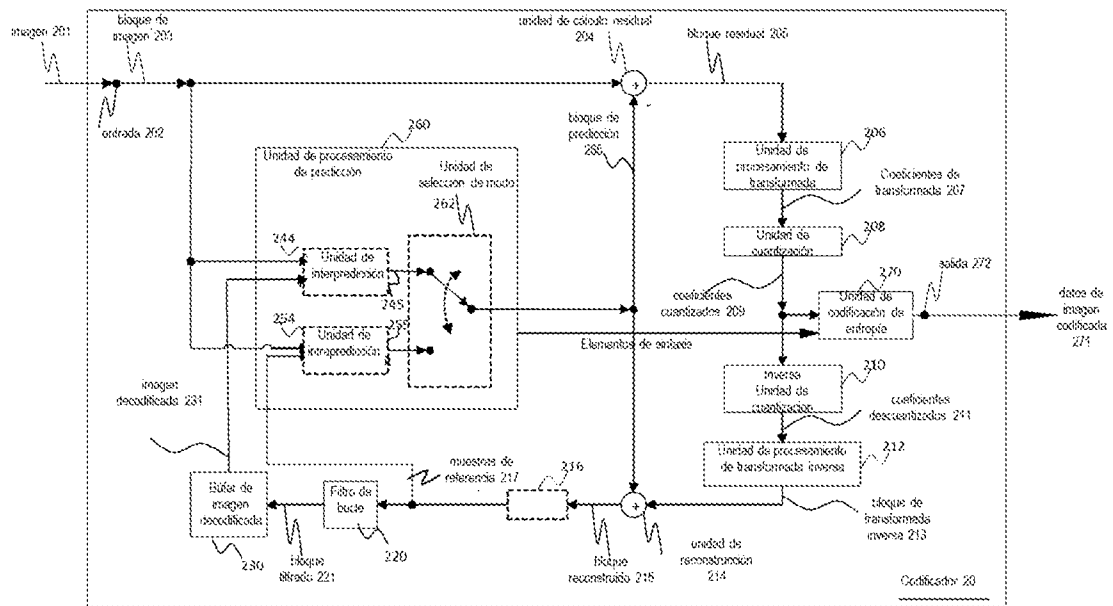


Fig. 2

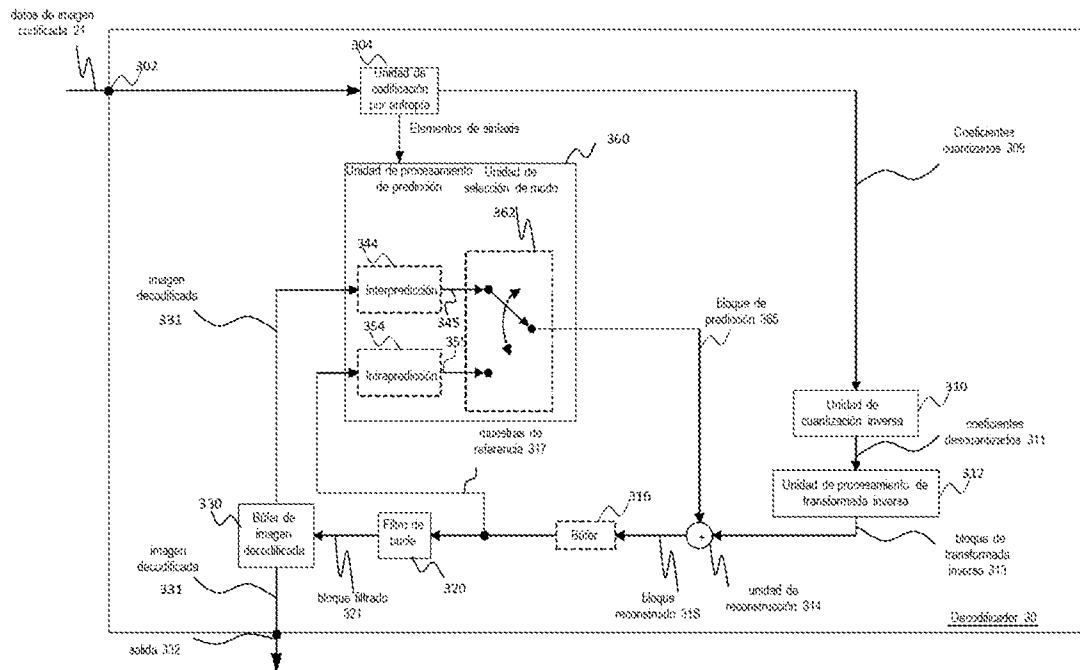


Fig. 3

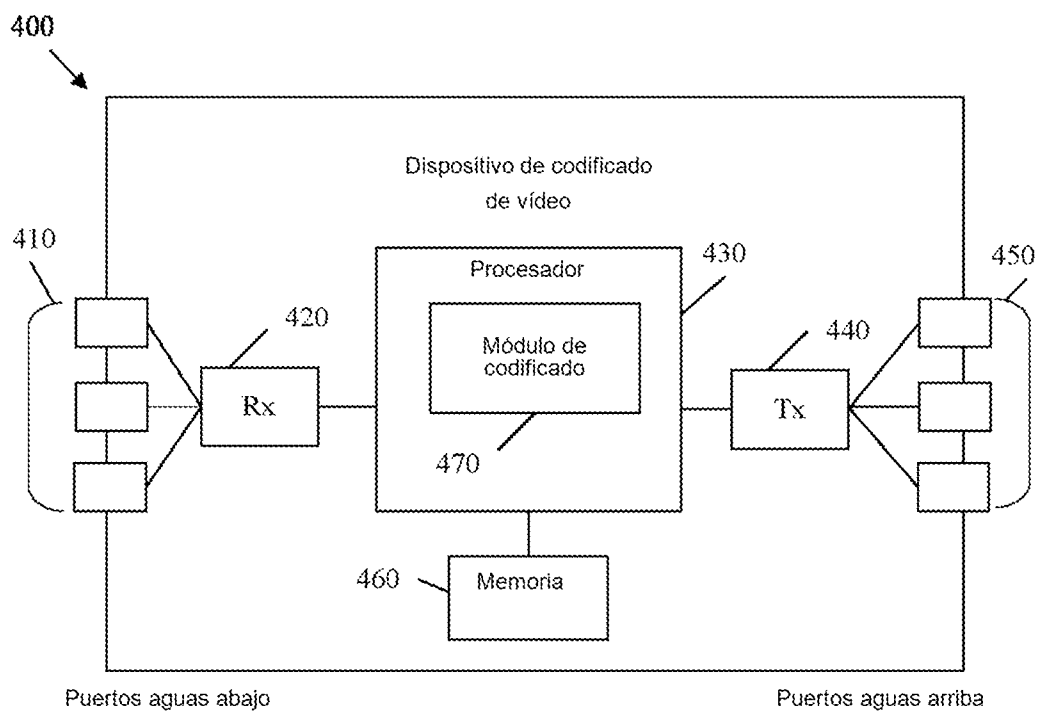


Fig. 4

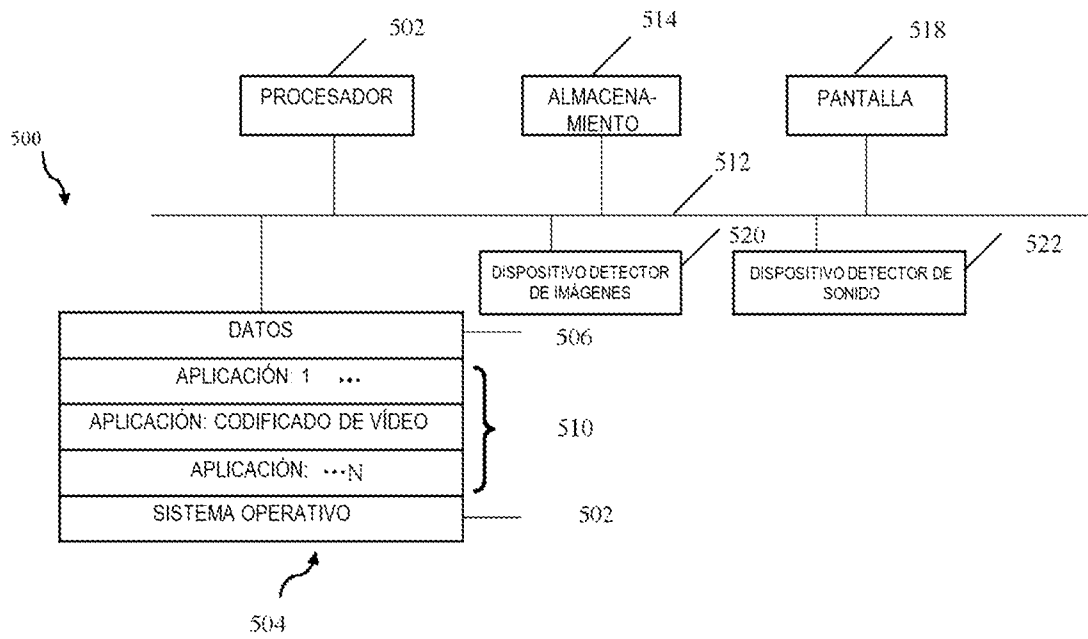


Fig. 5

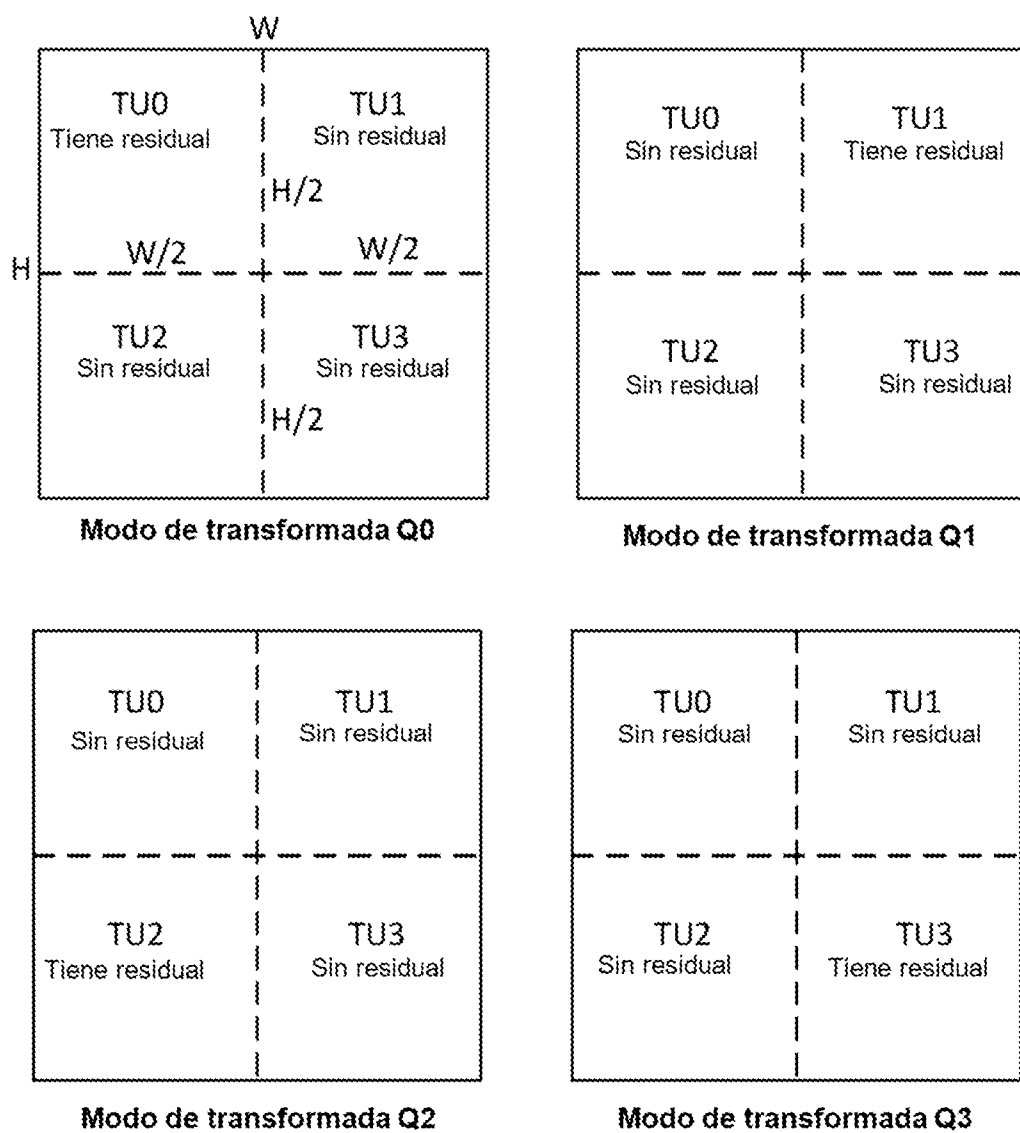


Fig. 6

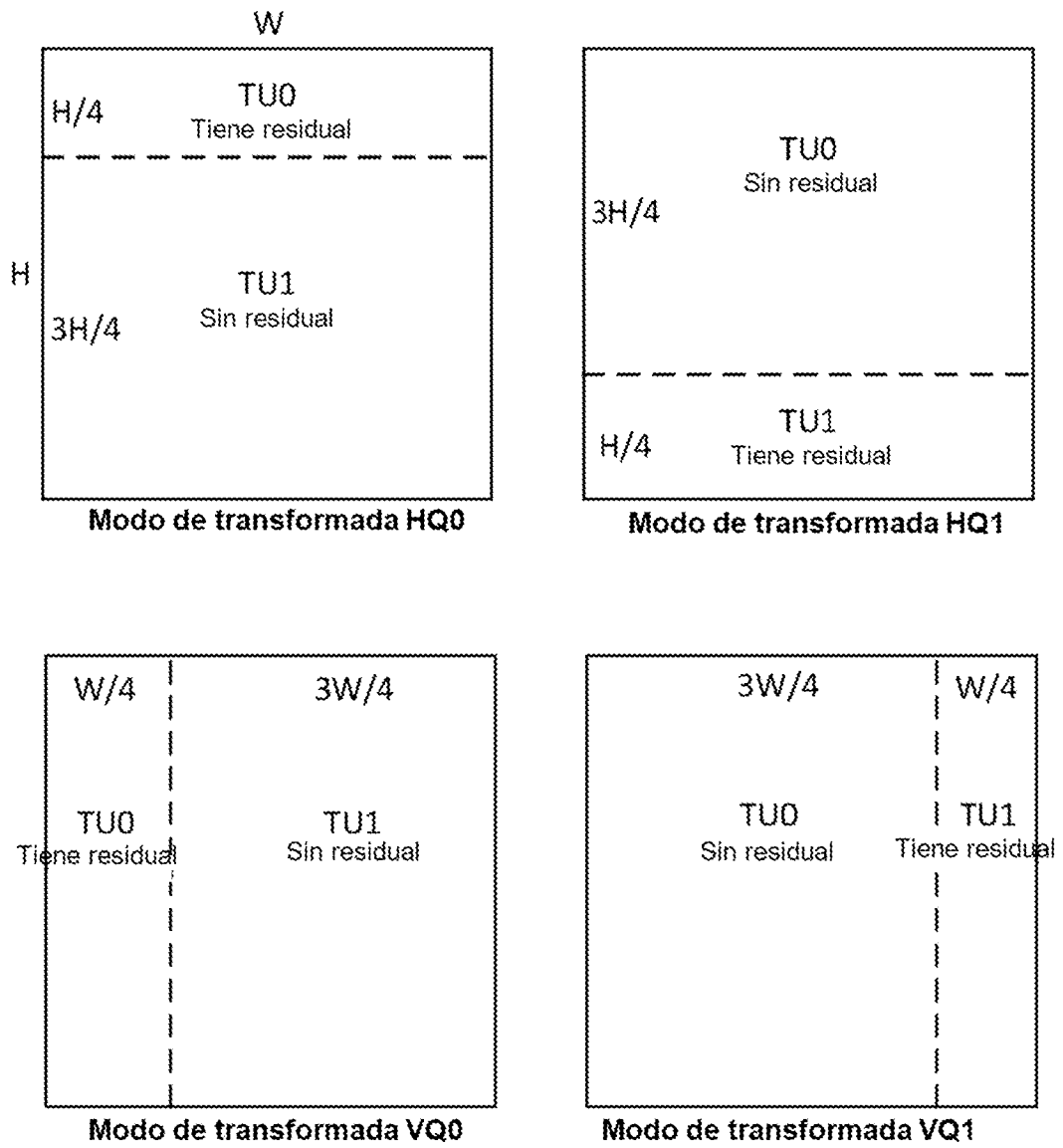


Fig. 7

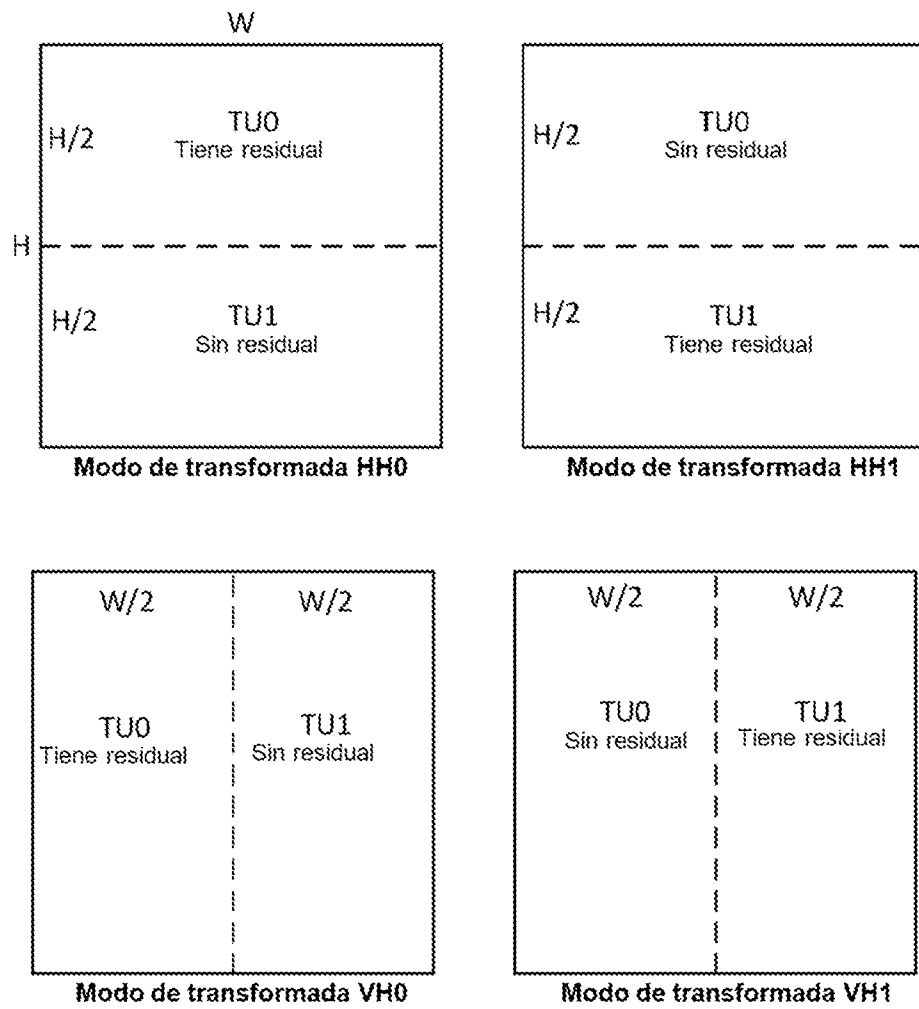


Fig. 8

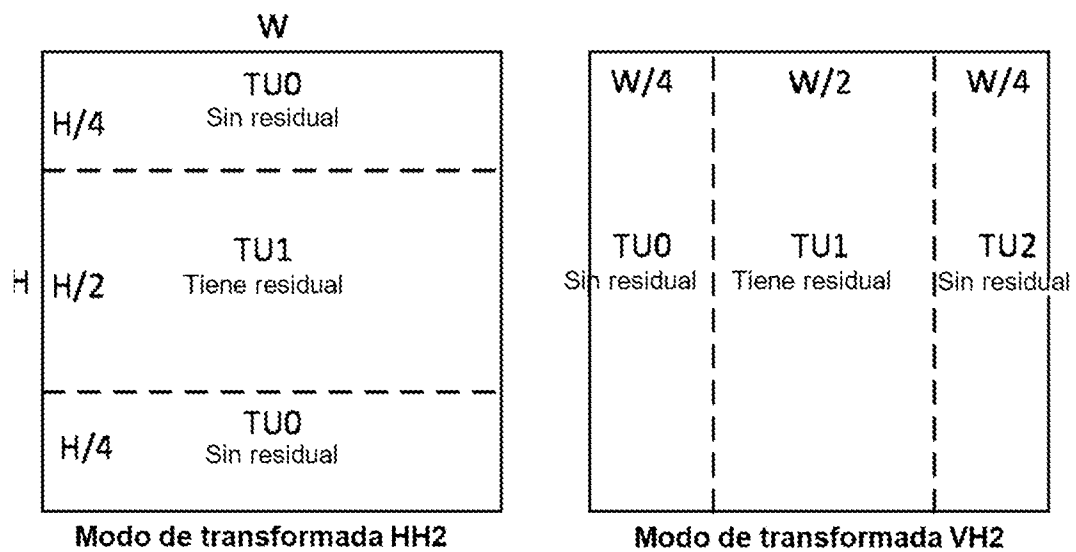


Fig. 9

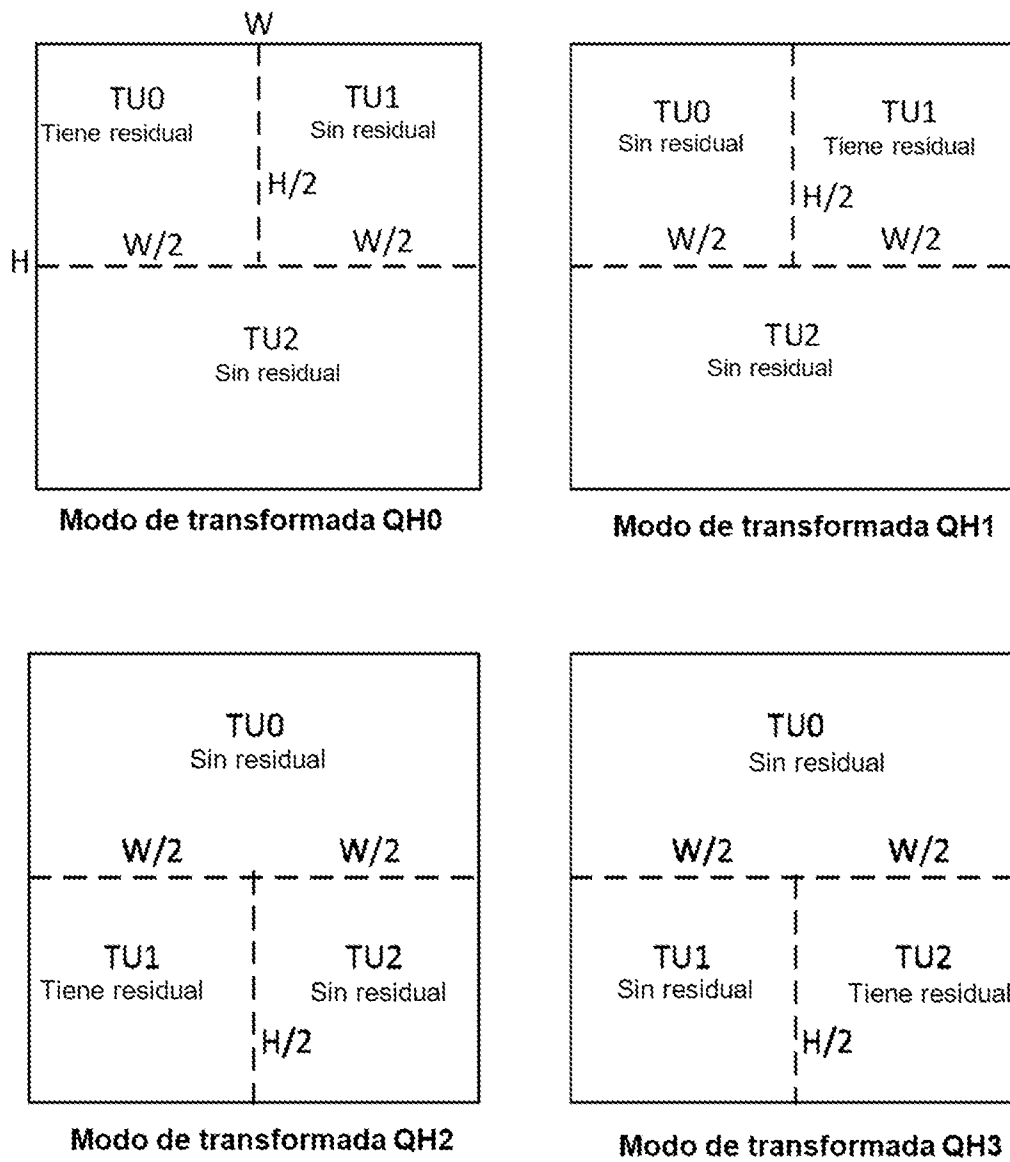


Fig. 10

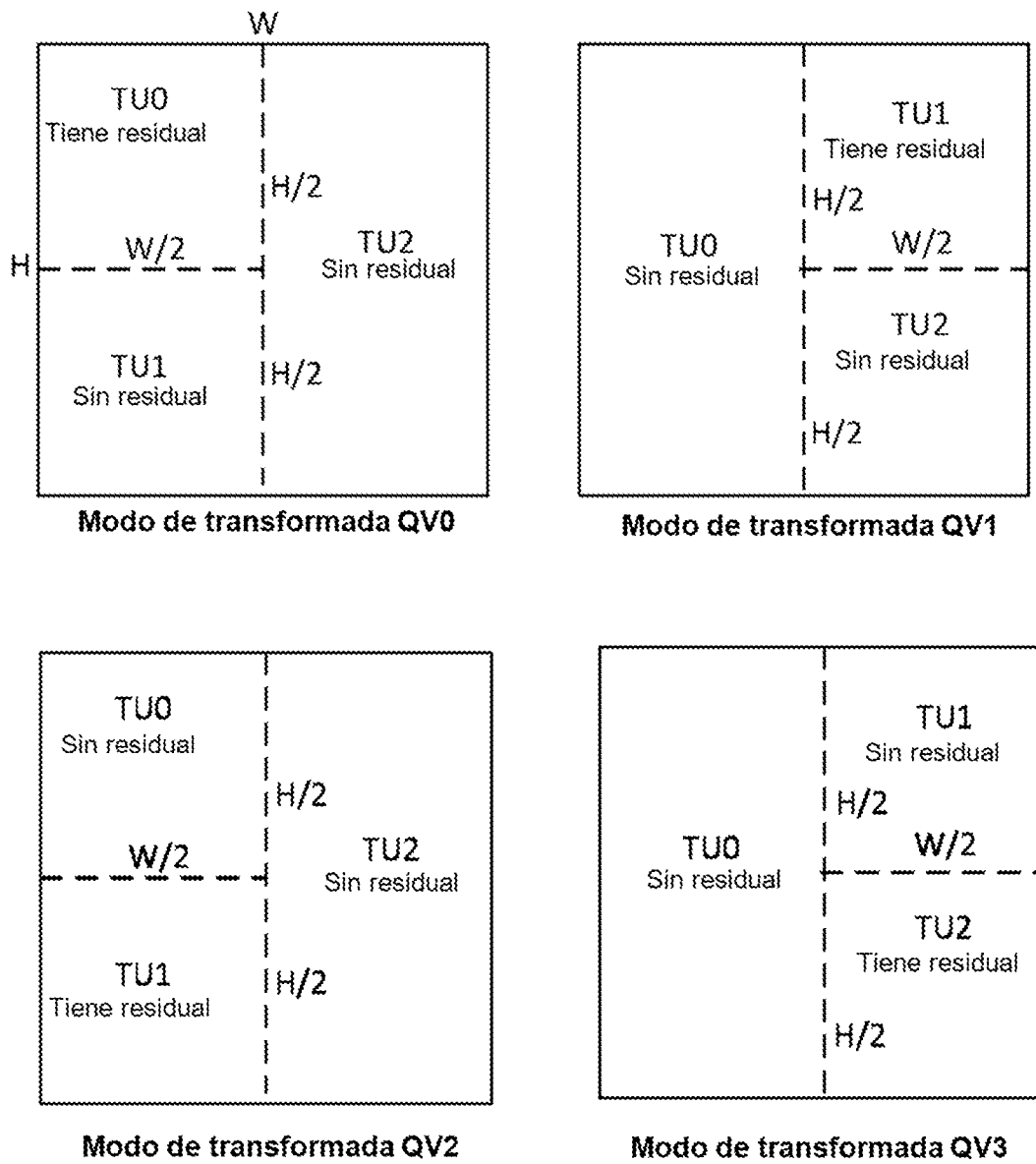


Fig. 11

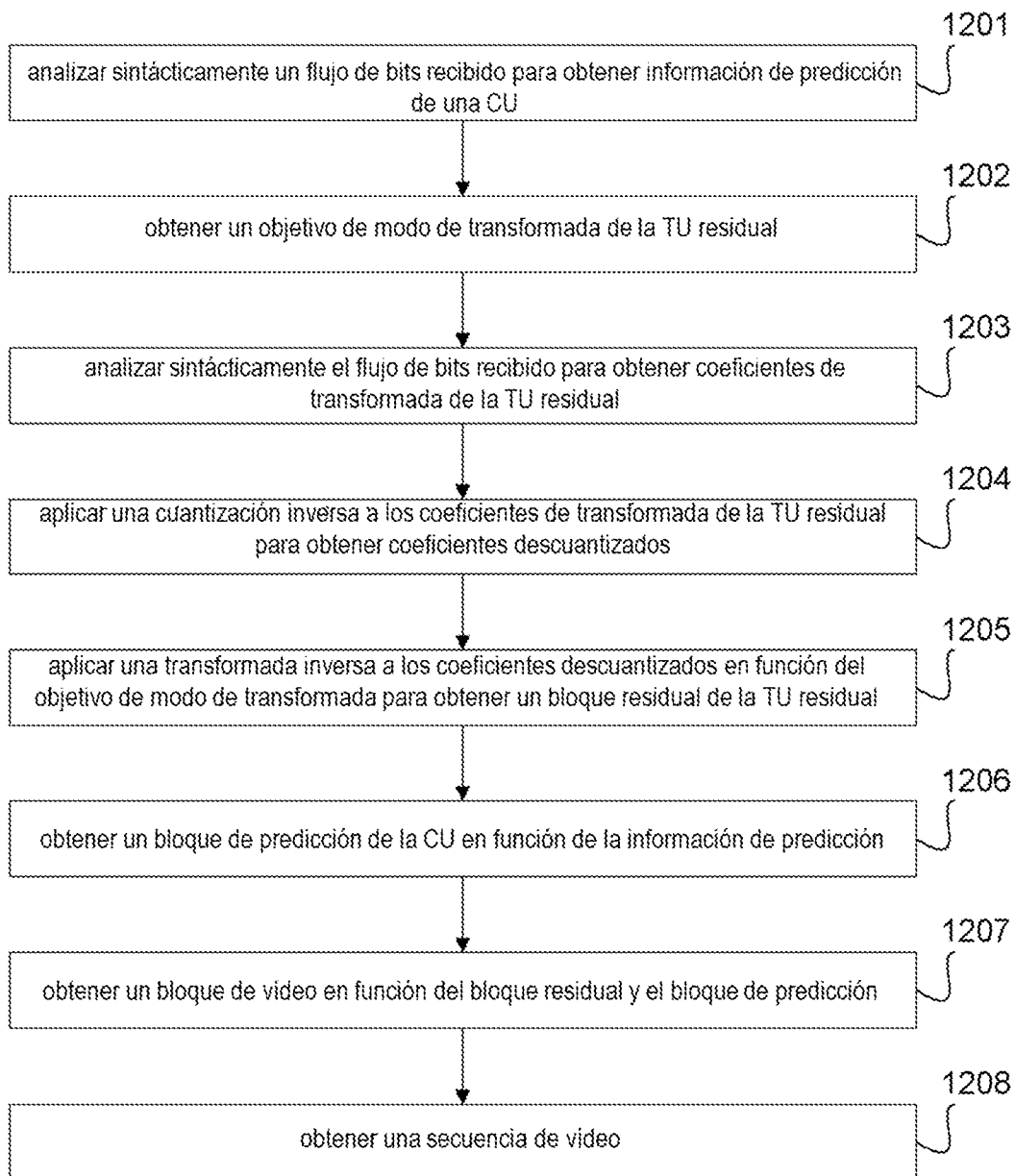


Fig. 12A

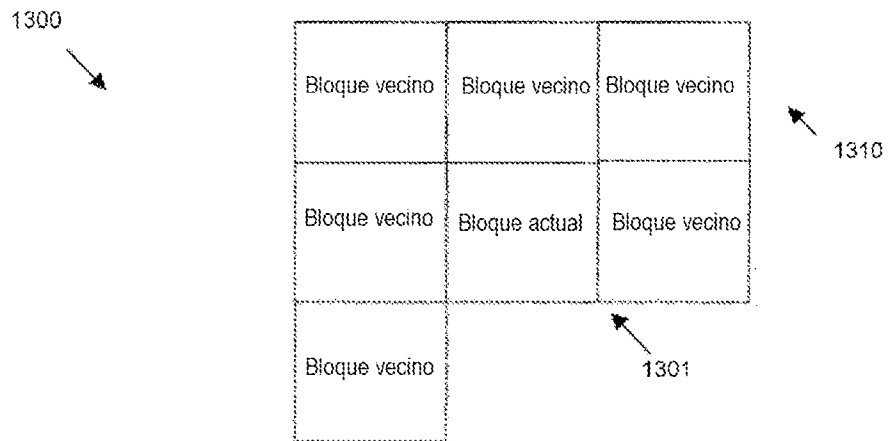


Fig. 13

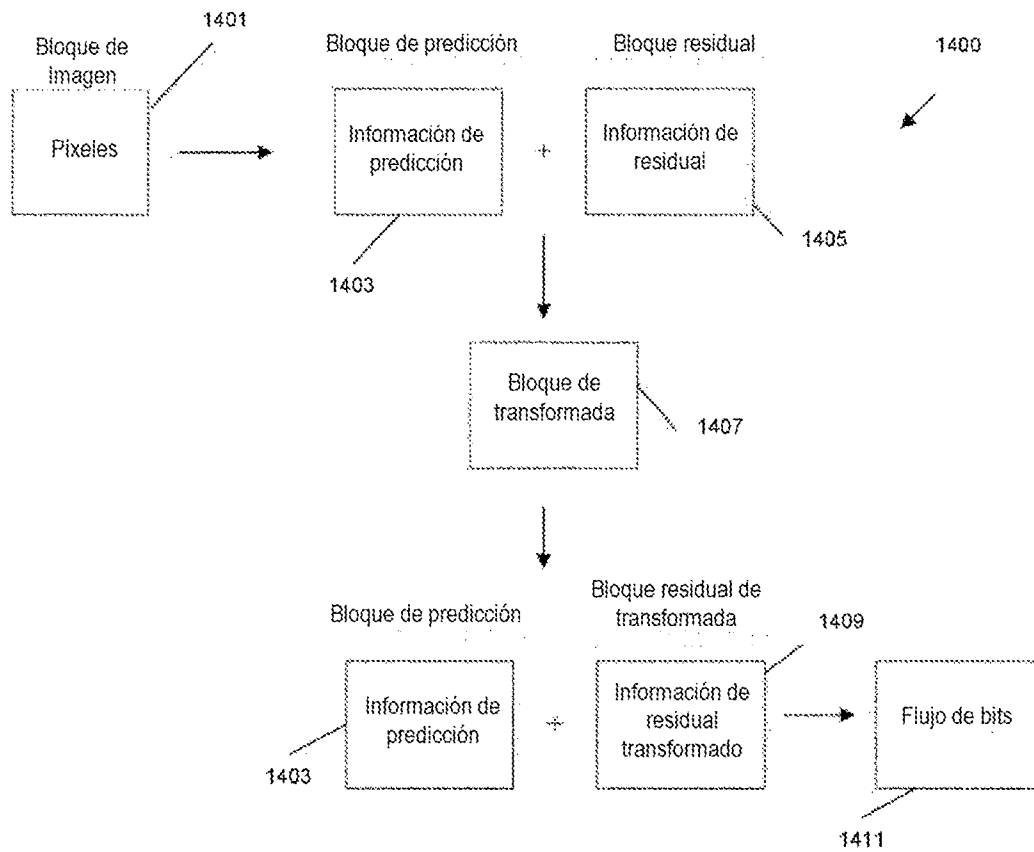


Fig. 14

1509

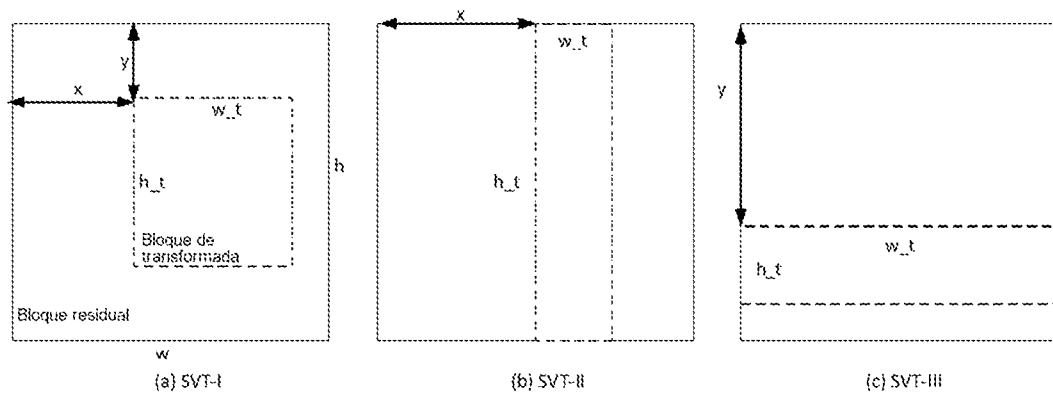


Fig. 15

Desplazamiento
de posición

1600

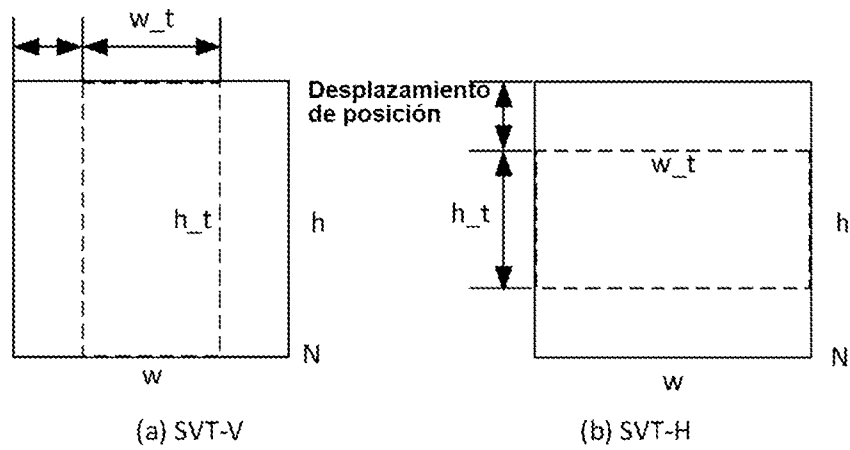


Fig. 16

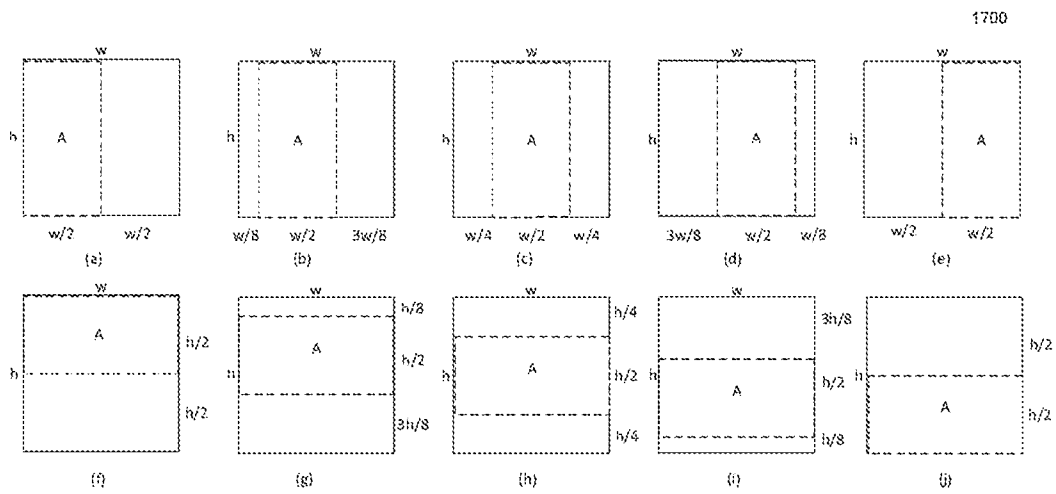


Fig. 17

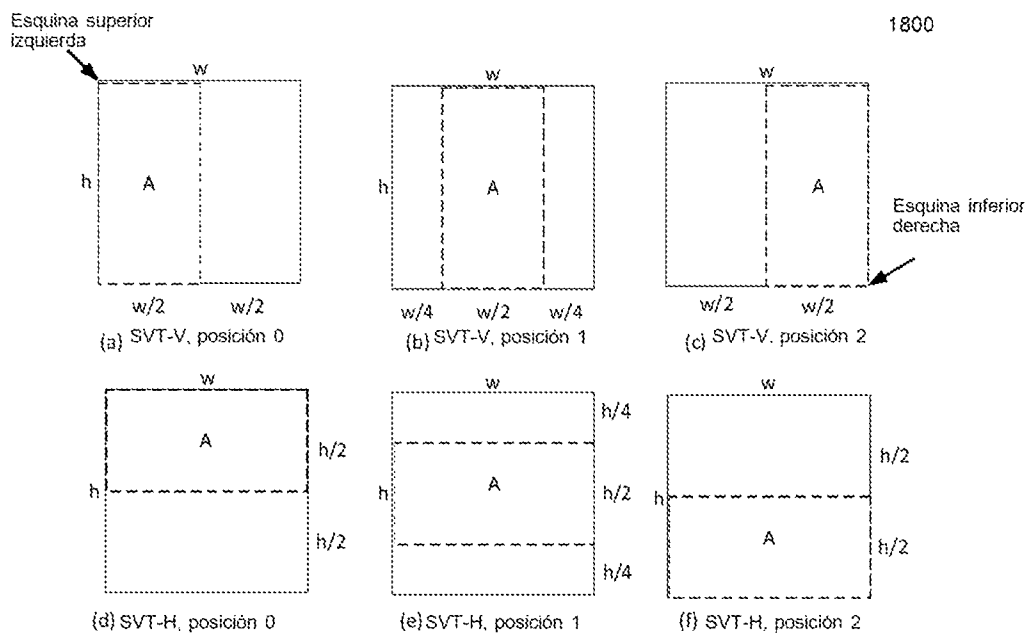


Fig. 18

1900

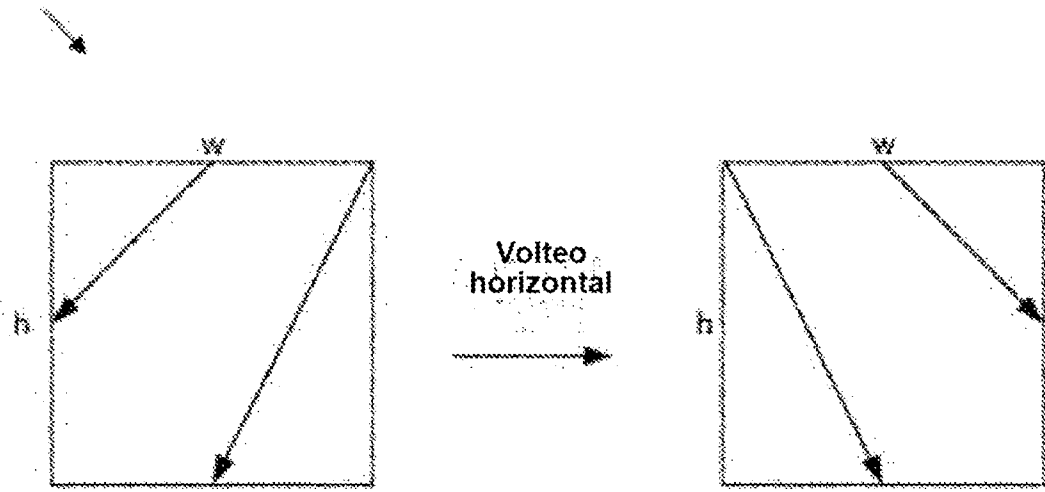


Fig. 19

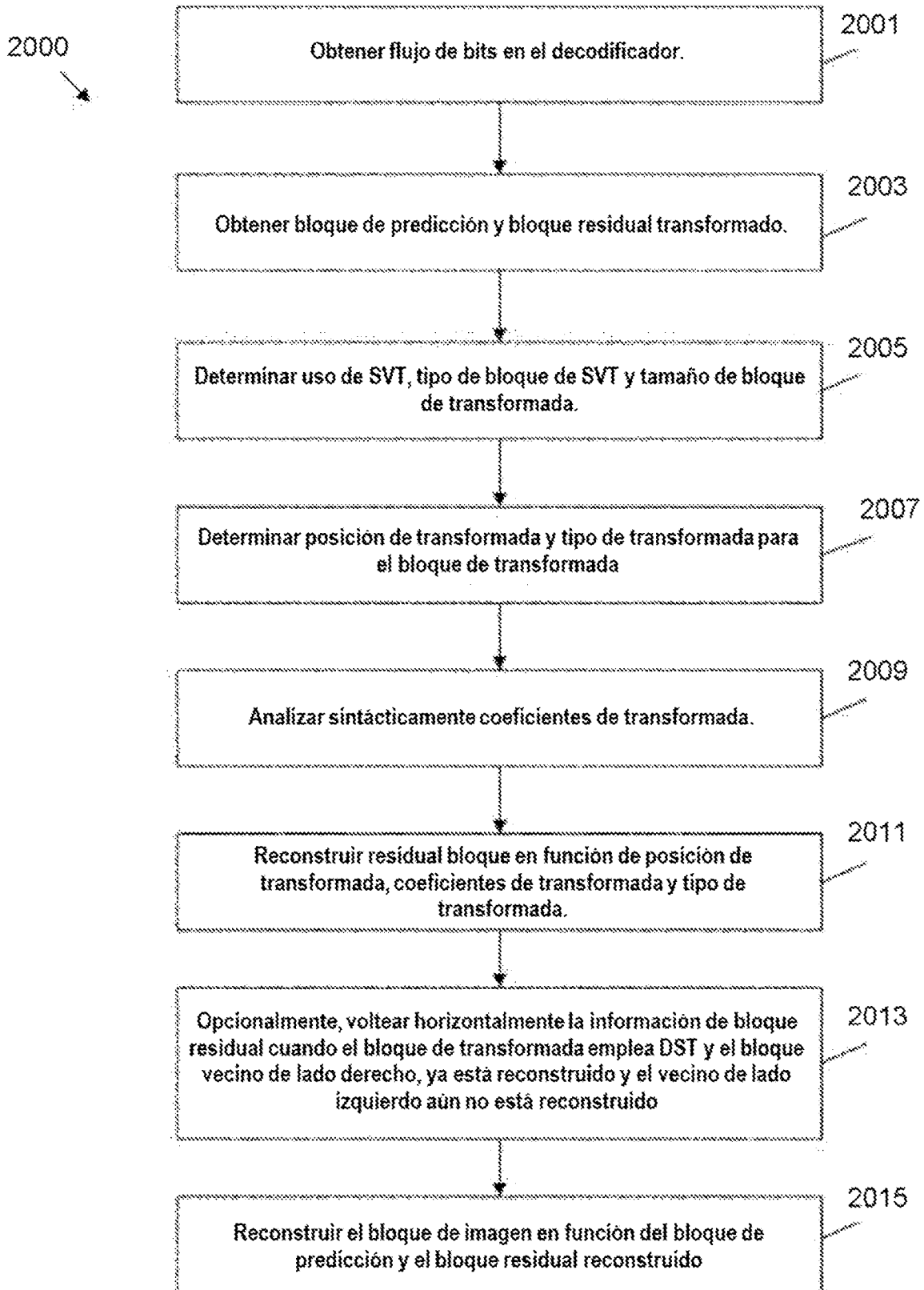


Fig. 20

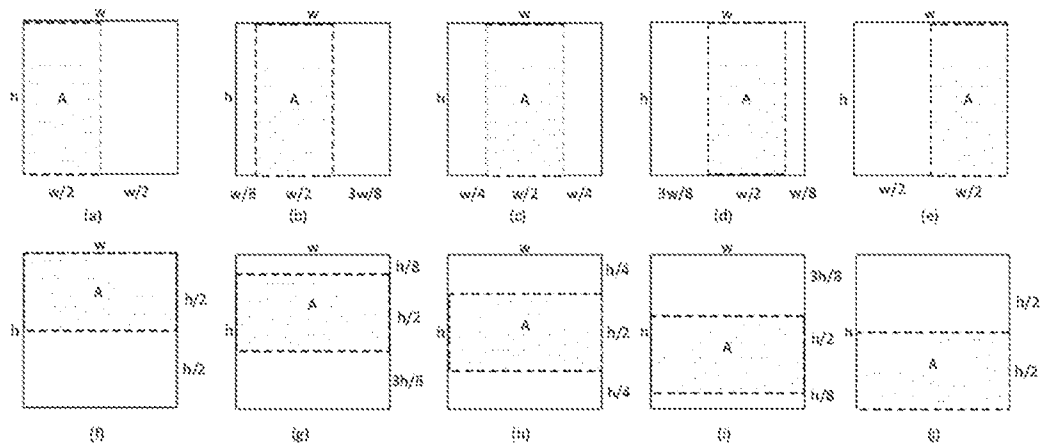


Fig. 21

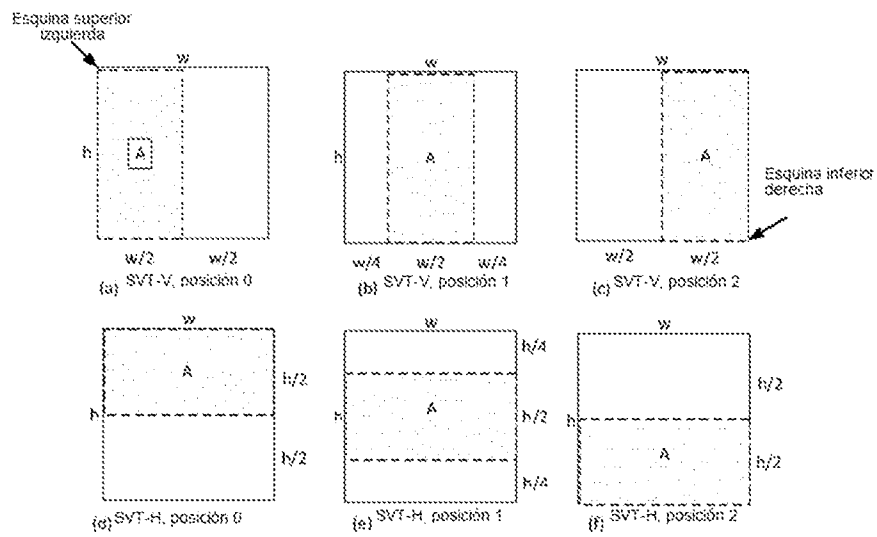
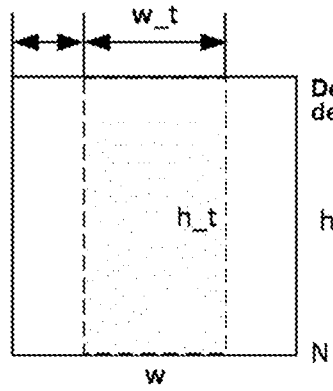


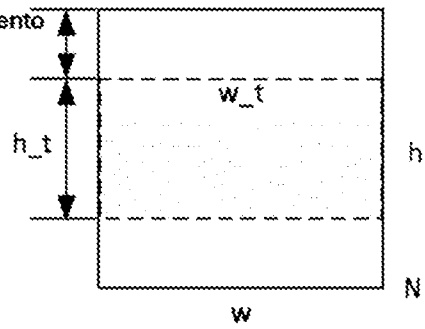
Fig. 22

Desplazamiento
de posición



(a) SVT-V

Desplazamiento
de posición



(b) SVT-H

Fig. 23

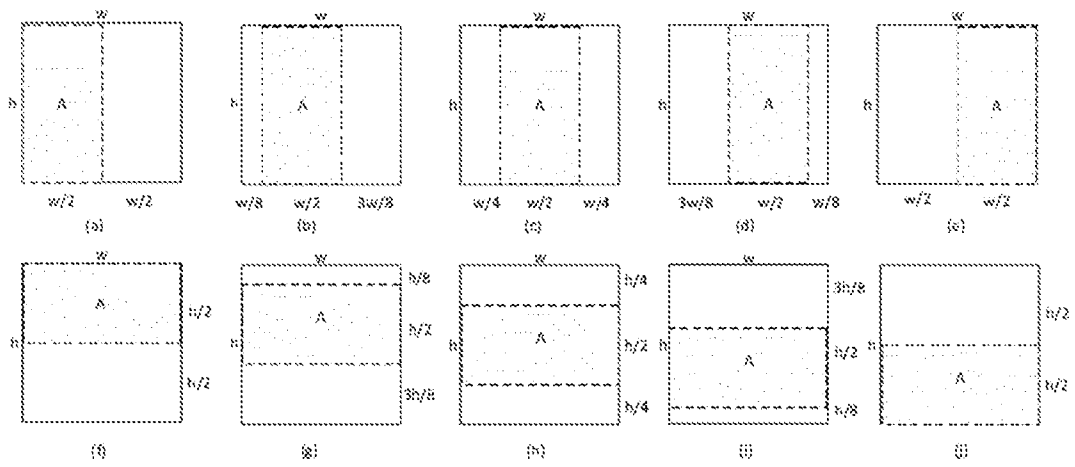


Fig. 24

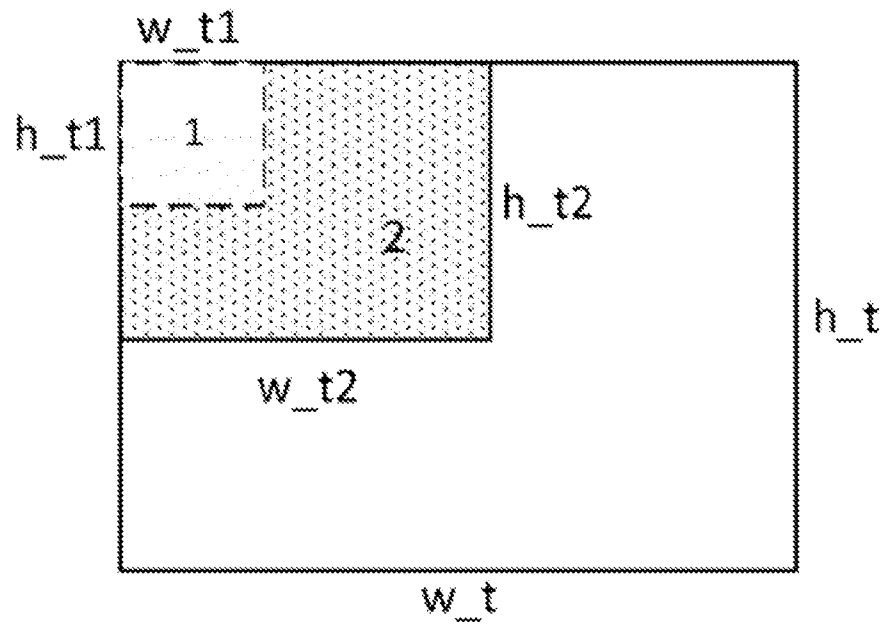


Fig. 25

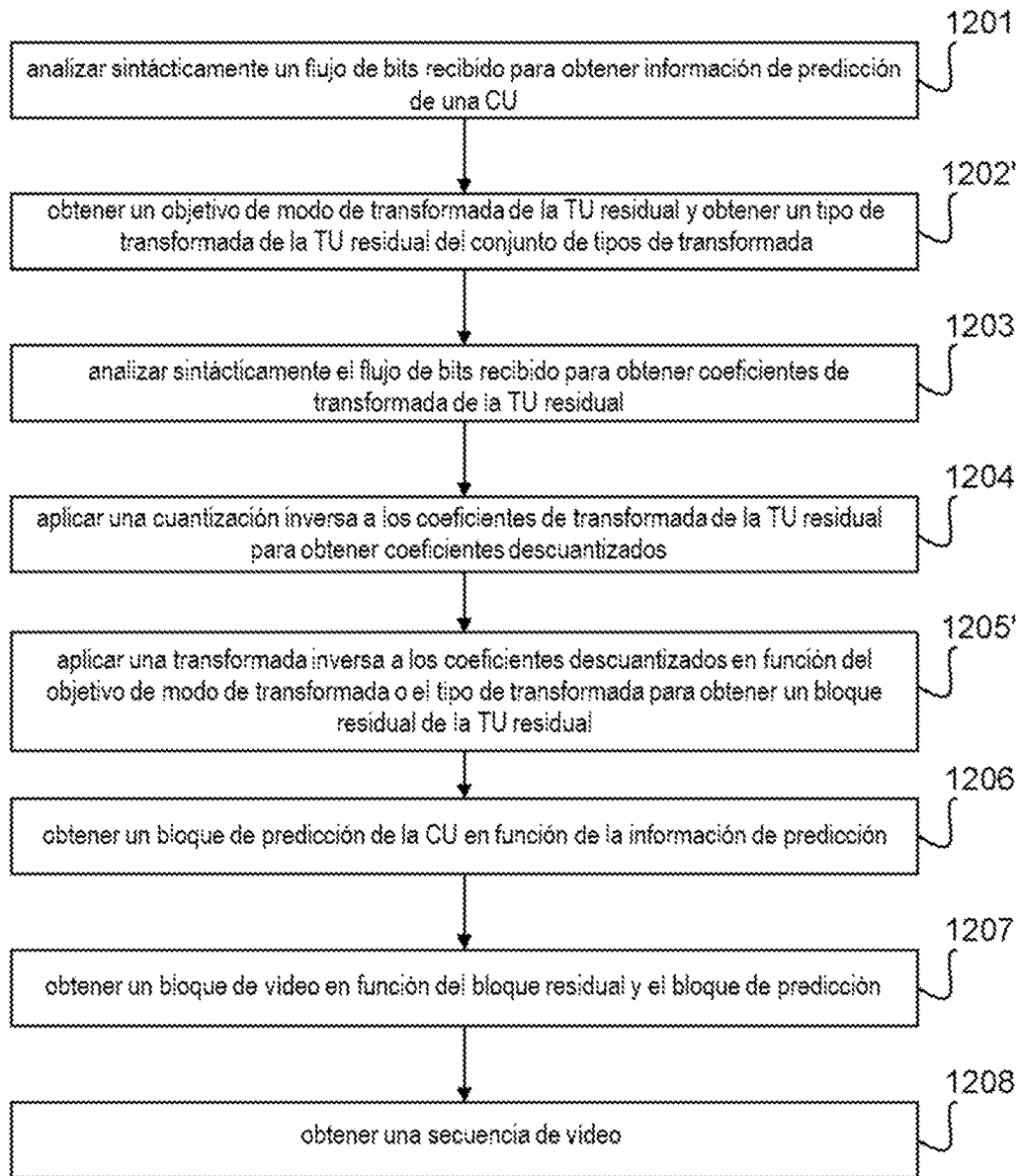


Fig. 12B