

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 794 619**

51 Int. Cl.:

H04N 19/176 (2014.01)
H04N 19/593 (2014.01)
H04N 19/11 (2014.01)
H04N 19/117 (2014.01)
H04N 19/463 (2014.01)
H04N 19/136 (2014.01)
H04N 19/182 (2014.01)
H04N 19/82 (2014.01)
H04N 19/59 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **17.06.2016** **PCT/US2016/038132**
- 87 Fecha y número de publicación internacional: **22.12.2016** **WO16205684**
- 96 Fecha de presentación y número de la solicitud europea: **17.06.2016** **E 16736954 (5)**
- 97 Fecha y número de publicación de la concesión europea: **26.02.2020** **EP 3311571**

54 Título: **Intrapredicción y codificación intramodo**

30 Prioridad:

18.06.2015 US 201562181744 P
14.07.2015 US 201562192310 P
16.06.2016 US 201615184051

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
18.11.2020

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
International IP Administration, 5775 Morehouse
Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

ZHAO, XIN;
CHEN, JIANLE;
ZHANG, LI y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 794 619 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

Intrapredicción y codificación intramodo

5 CAMPO TÉCNICO

[0001] Esta divulgación se refiere a la codificación y a la decodificación de vídeo.

ANTECEDENTES

10 [0002] Las capacidades de vídeo digital se pueden incorporar en una amplia gama de dispositivos, que incluye televisores digitales, sistemas de radiodifusión digital directa, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, tabletas electrónicas, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), la norma de Codificación de Vídeo de Alta Eficacia (HEVC) y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital con más eficacia implementando dichas técnicas de compresión de vídeo.

25 [0003] Las técnicas de compresión de vídeo realizan predicción espacial (intraimagen) y/o predicción temporal (interimagen) para reducir o eliminar la redundancia intrínseca a las secuencias de vídeo. En la codificación de vídeo basada en bloques, un fragmento de vídeo (es decir, un cuadro de vídeo o una porción de un cuadro de vídeo) se puede dividir en bloques de vídeo. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar predicción espacial con respecto a muestras de referencia de bloques vecinos en la misma imagen o predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes se pueden denominar cuadros, y las imágenes de referencia se pueden denominar cuadros de referencia.

35 [0004] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y de acuerdo con datos residuales. Para una mayor compresión, los datos residuales se pueden transformar desde el dominio de píxel hasta un dominio de transformada, dando como resultado coeficientes residuales que, a continuación, se pueden cuantificar. Los coeficientes cuantificados, dispuestos inicialmente en una matriz bidimensional, se pueden explorar para generar un vector unidimensional de coeficientes, y se puede aplicar codificación por entropía para lograr incluso más compresión.

45 [0005] La solicitud de patente internacional con número de publicación WO 2013/067344 A2 describe formas de determinar los modos más probables (MPM) para bloques de intrapredicción de datos de vídeo.

50 [0006] El documento de TD Chuang *et al.* "CE6b: Intra prediction mode coding", 7. Reunión de JCT-VC; 98. Reunión de MPEG, describe la determinación de MPM para la intrapredicción. Propone añadir tres MPM de acuerdo con los modos de intrapredicción de los bloques izquierdo y superior.

[0007] La solicitud de patente internacional con número de publicación WO 2013/000324 A1 describe procedimientos de codificación en modo de intrapredicción, incluidos procedimientos que usan MPM.

55 [0008] El capítulo 4: Intrapredicción en HEVC en el documento "High Efficiency Video Coding (HEVC)" por Jani Laiema *et al.* describe el marco de intrapredicción usado en HEVC.

BREVE EXPLICACIÓN

60 [0009] El alcance de protección está definido por las reivindicaciones independientes. Las características opcionales están incluidas en las reivindicaciones dependientes.

[0010] En general, esta divulgación describe técnicas para la intrapredicción y la codificación en modo de intrapredicción, que se pueden usar en el contexto de códecs de vídeo avanzados, tales como ampliaciones de la norma de Codificación de Vídeo de Alta Eficacia (HEVC), o la próxima generación de normas de codificación de vídeo.

65

[0011] De acuerdo con un aspecto de la invención reivindicada, hay un procedimiento para codificar un bloque actual de datos de vídeo. El procedimiento comprende obtener M modos más probables (MPM) para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior. Obtener los M modos más probables comprende al menos uno de: (i) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda; definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda, o (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior; definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior; codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo; codificar el índice indicado del índice MPM o el índice no MPM, donde el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y codificar el bloque de datos de vídeo en base al modo de intrapredicción seleccionado.

[0012] En otro aspecto, se proporciona un procedimiento para descodificar un bloque actual de datos de vídeo. El procedimiento comprende obtener M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior. Obtener los M modos más probables comprende al menos uno de: (i) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda; definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda, o (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior; definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior; descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo; descodificar el índice indicado del índice MPM o el índice no MPM, donde el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y reconstruir el bloque actual de datos de vídeo en base al modo de intrapredicción seleccionado.

[0013] En otro aspecto, se proporciona un dispositivo para codificar datos de vídeo. El dispositivo comprende una memoria configurada para almacenar los datos de vídeo; y uno o más procesadores configurados para: obtener M MPM para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y el uno o más procesadores están configurados de modo que, como parte de la obtención de los M modos más probables, el uno o más procesadores realizan al menos uno de: (i) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda; definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda, o (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior; definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior; codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo; codificar el índice indicado del índice de modo más probable o el índice de modo que no es más probable, donde el índice de modo más probable indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y codificar el bloque actual de datos de vídeo en base al modo de intrapredicción seleccionado.

[0014] En otro aspecto se proporciona un dispositivo de descodificación de datos de vídeo, que puede incluir una memoria configurada para almacenar los datos de vídeo; y uno o más procesadores configurados para: obtener M MPM para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila

vecina superior, y el uno o más procesadores están configurados de modo que, como parte de la obtención de los M modos más probables, el uno o más procesadores realizan al menos uno de: (i) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda; definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda, o (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior; definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior; descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo; descodificar el índice indicado del índice de modo más probable o el índice de modo que no es más probable, donde el índice de modo más probable indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y reconstruir el bloque actual de datos de vídeo en base al modo de intrapredicción seleccionado.

[0015] En otro aspecto, se proporciona un medio legible por ordenador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores, hacen que el uno o más procesadores: obtengan M MPM para la intrapredicción de un bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y como parte de hacer que el uno o más procesadores obtengan los M modos más probables, las instrucciones, cuando son ejecutadas por el uno o más procesadores, hacen que el uno o más procesadores realicen al menos uno de: (i) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda; definir un modo de intrapredicción representativo para la columna vecina izquierda y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda, o (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, los modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior; definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior; codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo; codificar el índice indicado del índice MPM o el índice no MPM, donde el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y codificar el bloque actual de datos de vídeo en base al modo de intrapredicción seleccionado.

[0016] Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0017]

La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo de ejemplo que puede utilizar las técnicas descritas en esta divulgación.

La FIG. 2 es un diagrama conceptual que ilustra un ejemplo de intrapredicción de un bloque de datos de vídeo.

La FIG. 3 es un diagrama conceptual que ilustra un ejemplo de modos de intrapredicción e índices de modo correspondientes.

La FIG. 4 es un diagrama conceptual que ilustra una técnica de ejemplo para generar una muestra de predicción para un bloque de datos de vídeo de acuerdo con un modo de intrapredicción planar.

La FIG. 5 es un diagrama conceptual que ilustra una técnica de ejemplo para generar una muestra de predicción para un bloque de datos de vídeo de acuerdo con un modo de intrapredicción angular.

La FIG. 6 es un diagrama de bloques que ilustra un codificador de vídeo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

La FIG. 7 es un diagrama de bloques que ilustra un descodificador de vídeo de ejemplo que puede implementar las técnicas descritas en esta divulgación.

La FIG. 8 es un diagrama conceptual que ilustra un ejemplo de modos de intrapredicción e índices de modo correspondientes de acuerdo con las técnicas de esta divulgación.

5 La FIG. 9 es un diagrama conceptual que ilustra ángulos de intrapredicción de ejemplo de acuerdo con las técnicas de esta divulgación.

La FIG. 10 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

10 La FIG. 11 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

15 La FIG. 12 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

La FIG. 13 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

20 La fig. 14 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

La FIG. 15 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

25 La FIG. 16 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

30 La FIG. 17 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

La FIG. 18 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

35 La FIG. 19 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

La FIG. 20 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

40 La FIG. 21 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

45 **[0018]** En general, esta divulgación describe técnicas para la intrapredicción y la codificación en modo de intrapredicción, que se pueden usar en el contexto de códecs de vídeo avanzados, tales como ampliaciones de la norma HEVC, o la próxima generación de normas de codificación de vídeo. Por ejemplo, esta divulgación describe técnicas para obtener, seleccionar y/o señalar los modos más probables (MPM) para la intrapredicción. Los ejemplos descritos en esta divulgación también incluyen técnicas para la intrapredicción usando un mayor número de modos angulares. Las técnicas para la intrapredicción usando un mayor número de modos angulares pueden incluir técnicas para aplicar un filtro de interpolación intra de N etapas, donde N es mayor que 2. La divulgación también describe técnicas que pueden permitir múltiples direcciones de intrapredicción por bloque, por ejemplo, direcciones respectivas para subbloques del bloque, que pueden no requerir señalización adicional de múltiples modos de intrapredicción.

55 **[0019]** La intrapredicción y la codificación intramodo son técnicas que se pueden usar en el contexto de códecs de vídeo avanzados, tales como ampliaciones de la norma de Codificación de Vídeo de Alta Eficacia (HEVC) o la próxima generación de normas de codificación de vídeo.

60 **[0020]** En la codificación intramodo en HEVC, para cada unidad de intrapredicción (PU), se señala un modo de intrapredicción seleccionado. Para seleccionar el modo de intrapredicción, primero se identifican tres intramodos, que se supone que tienen mayor probabilidad de ser realmente seleccionados, a saber, los modos más probables (MPM). En HEVC, hay 35 modos para la intrapredicción de un bloque de luma, que incluyen un modo de intrapredicción planar, un modo de intrapredicción DC y 33 direcciones de predicción diferentes asociadas a modos de intrapredicción angulares, respectivamente.

65

[0021] Los 35 ángulos de predicción fijos del diseño existente de intrapredicción en HEVC pueden ser ineficaces para capturar distribuciones de dirección de borde muy flexibles. Sin embargo, la aplicación de más ángulos de predicción puede dar como resultado un aumento no deseado de la complejidad del codificador. Por ejemplo, una ampliación directa a 65 ángulos de predicción en base al software de referencia de HEVC puede requerir un número aproximadamente duplicado de verificaciones SATD. Además, la eficacia de la codificación intramodo actual, usando tres MPM, en HEVC puede ser limitada ya que puede no corresponder exactamente con la distribución de probabilidad real de todos los modos de intrapredicción disponibles.

[0022] Esta divulgación describe técnicas que pueden remediar una o más de estas deficiencias en HEVC. Por ejemplo, de acuerdo con algunas técnicas de esta divulgación, un codificador de vídeo puede obtener más de tres MPM. En algunos de dichos ejemplos, el codificador de vídeo puede usar modelado de contexto para decodificar uno o más *bins* de un índice MPM que indica cuál de los MPM es un MPM seleccionado para una PU actual. En algunos ejemplos en los que el codificador de vídeo obtiene más de tres MPM, el codificador de vídeo puede definir, entre los MPM, un modo de intrapredicción representativo para la columna vecina izquierda y usar el modo de intrapredicción representativo para la columna vecina izquierda como MPM para la columna vecina izquierda y/o definir un modo de intrapredicción representativo para la fila vecina superior y usar el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior. Además, en algunos ejemplos en los que el codificador de vídeo obtiene más de tres MPM, el codificador de vídeo puede seleccionar uno o más MPM angulares adicionales en función de la similitud con un modo angular ya presente entre los MPM. En este ejemplo, la similitud se determina en base a al menos una de diferencias de índice de modo de intrapredicción o diferencias de ángulo de intrapredicción. Además, en algunas técnicas de ejemplo de esta divulgación, un índice no MPM puede codificarse en el flujo de bits como una palabra de código más corta que $\lceil \log_2 N \rceil$ bits si el índice no MPM cumple un criterio y se codifica como un código de longitud fija con $\lceil \log_2 N \rceil$ bits; de lo contrario, hay un total de N valores disponibles del índice no MPM.

[0023] En algunos ejemplos de esta divulgación donde el codificador de vídeo puede usar más de 33 modos de intrapredicción angulares, el codificador de vídeo puede usar un filtro de interpolación en muestras reconstruidas vecinas en las cuales la interpolación tiene una precisión de 1/32 pel. En algunos ejemplos de esta divulgación, el codificador de vídeo puede calcular un valor de una muestra respectiva aplicando un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas para interpolar un valor en la posición fraccionada determinada, donde N es mayor que 2.

[0024] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 de ejemplo que puede utilizar las técnicas de esta divulgación. Como se usa en el presente documento, el término "codificador de vídeo" se refiere genéricamente tanto a codificadores de vídeo como a decodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o la decodificación de vídeo.

[0025] Como se muestra en la FIG. 1, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12 y un dispositivo de destino 14. El dispositivo de origen 12 genera datos de vídeo codificados. Por consiguiente, el dispositivo de origen 12 se puede denominar dispositivo de codificación de vídeo o aparato de codificación de vídeo. El dispositivo de destino 14 puede decodificar los datos de vídeo codificados generados por el dispositivo de origen 12. Por consiguiente, el dispositivo de destino 14 se puede denominar dispositivo de decodificación de vídeo o aparato de decodificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender una amplia gama de dispositivos, que incluye ordenadores de escritorio, dispositivos informáticos móviles, ordenadores tipo *notebook* (por ejemplo, portátiles), tabletas electrónicas, decodificadores multimedia, aparatos telefónicos tales como los denominados teléfonos "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, ordenadores para vehículos o similares.

[0026] El dispositivo de destino 14 puede recibir datos de vídeo codificados desde el dispositivo de origen 12 por medio de un canal 16. El canal 16 puede comprender uno o más medios o dispositivos capaces de transferir los datos de vídeo codificados desde el dispositivo de origen 12 hasta el dispositivo de destino 14. En un ejemplo, el canal 16 puede comprender uno o más medios de comunicación que permiten al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. En este ejemplo, el dispositivo de origen 12 puede modular los datos de vídeo codificados de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y puede transmitir los datos de vídeo modulados al dispositivo de destino 14. Los uno o más medios de comunicación pueden incluir medios de comunicación inalámbrica y/o alámbrica, tales como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. Los uno o más medios de comunicación pueden formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global (por ejemplo, Internet). Los uno o más medios de comunicación pueden incluir encaminadores, conmutadores, estaciones base u otros equipos que faciliten la comunicación desde el dispositivo de origen 12 hasta el dispositivo de destino 14.

[0027] En algunos ejemplos, el canal 16 puede incluir un medio de almacenamiento que almacena datos de vídeo codificados generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder al

medio de almacenamiento, por ejemplo, por medio de acceso a disco o acceso a tarjeta. El medio de almacenamiento puede incluir una variedad de medios de almacenamiento de datos de acceso local, tales como discos Blu-ray, DVD, CD-ROM, memoria flash u otros medios de almacenamiento digital adecuados para almacenar datos de vídeo codificados.

[0028] En algunos ejemplos, el canal 16 puede incluir un servidor de archivos u otro dispositivo de almacenamiento intermedio que almacena datos de vídeo codificados generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder a datos de vídeo codificados almacenados en el servidor de archivos o en otro dispositivo de almacenamiento intermedio por medio de transmisión continua o descarga. El servidor de archivos puede ser un tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir los datos de vídeo codificados al dispositivo de destino 14. Ejemplos de servidores de archivos incluyen servidores web (por ejemplo, para un sitio web), servidores de protocolo de transferencia de archivos (FTP), dispositivos de almacenamiento conectado a red (NAS) y unidades de disco locales.

[0029] El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de una conexión de datos estándar, tal como una conexión a Internet. Ejemplos de tipos de conexiones de datos pueden incluir canales inalámbricos (por ejemplo, conexiones Wi-Fi), conexiones alámbricas (por ejemplo, línea de abonado digital (DSL), módem por cable, etc.), o combinaciones de ambas cosas que sean adecuadas para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el servidor de archivos puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas cosas.

[0030] Las técnicas de esta divulgación no están limitadas a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como soporte de una variedad de aplicaciones multimedia, tales como radiodifusiones de televisión por aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, por medio de Internet, codificación de datos de vídeo para su almacenamiento en un medio de almacenamiento de datos, decodificación de datos de vídeo almacenados en un medio de almacenamiento de datos u otras aplicaciones. En algunos ejemplos, el sistema de codificación de vídeo 10 puede estar configurado para admitir transmisión de vídeo unidireccional o bidireccional para admitir aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0031] La FIG. 1 es simplemente un ejemplo y las técnicas de esta divulgación se pueden aplicar a configuraciones de codificación de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no incluyan necesariamente ninguna comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, los datos se recuperan de una memoria local, se emiten en continuo por una red, o similares. 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 una memoria. En muchos ejemplos, la codificación y la decodificación se realizan mediante dispositivos que no se comunican entre sí, sino que simplemente codifican datos en una memoria y/o recuperan y decodifican datos de una memoria.

[0032] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos ejemplos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. La fuente de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo que contenga datos de vídeo previamente capturados, una interfaz de suministro de vídeo para recibir datos de vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de vídeo, o una combinación de dichas fuentes de datos de vídeo.

[0033] El codificador de vídeo 20 puede codificar datos de vídeo procedentes de la fuente de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 transmite directamente los datos de vídeo codificados al dispositivo de destino 14 por medio de la interfaz de salida 22. En otros ejemplos, los datos de vídeo codificados también pueden almacenarse en un medio de almacenamiento o en un servidor de archivos para su acceso posterior mediante el dispositivo de destino 14 para su decodificación y/o su reproducción.

[0034] En el ejemplo de la FIG. 1, el dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos ejemplos, la interfaz de entrada 28 incluye un receptor y/o un módem. La interfaz de entrada 28 puede recibir datos de vídeo codificados a través del canal 16. El decodificador de vídeo 30 puede decodificar datos de vídeo codificados. El dispositivo de visualización 32 puede mostrar los datos de vídeo decodificados. El dispositivo de visualización 32 puede estar integrado en, o ser externo a, el dispositivo de destino 14. El dispositivo de visualización 32 puede comprender una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0035] Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden implementarse como cualquiera de una variedad de circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables *in situ* (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 medio de almacenamiento legible por

ordenador no transitorio adecuado, y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cualquiera de lo anterior (incluyendo hardware, software, una combinación de hardware y software, etc.) puede considerarse como uno o más procesadores. Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 se pueden incluir en uno o más codificadores o decodificadores, de los que cualquiera se puede integrar como parte de un codificador/decodificador (CÓDEC) combinado en un dispositivo respectivo. Un aparato que incluye un codificador de vídeo 20 y/o un decodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

[0036] La fuente de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado previamente y/o una interfaz de suministro de vídeo para recibir vídeo desde un proveedor de contenidos de vídeo. Como otra alternativa, la fuente de vídeo 18 puede generar datos basados en gráficos de ordenador como fuente de vídeo, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, si la fuente de vídeo 18 es una cámara de vídeo, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los llamados teléfonos con cámara o videoteléfonos. Sin embargo, como se menciona anteriormente, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o alámbricas. En cada caso, el codificador de vídeo 20 puede codificar el vídeo capturado, precapturado o generado por ordenador. Después, la información de vídeo codificada se puede emitir por la interfaz de salida 22 al canal 16.

[0037] Muchas de las técnicas descritas en esta divulgación pueden realizarse tanto por el codificador de vídeo 20 como por el decodificador de vídeo 30. Por lo tanto, para facilitar la explicación, las técnicas pueden describirse con respecto a un codificador de vídeo, que puede ser un codificador de vídeo y/o un decodificador de vídeo, tal como el codificador de vídeo 20 y el decodificador de vídeo 30. Esta divulgación se puede referir, en general, al codificador de vídeo 20 que "señaliza" determinada información a otro dispositivo, tal como el decodificador de vídeo 30. El término "señalización" puede referirse, en general, a la comunicación de elementos sintácticos y/o a otros datos usados para decodificar los datos de vídeo comprimidos. Dicha comunicación se puede producir en tiempo real o casi real. De forma alternativa, dicha comunicación puede producirse durante un periodo de tiempo, tal como podría producirse cuando se almacenan elementos sintácticos en un medio de almacenamiento legible por ordenador en un flujo de bits codificado en el momento de la codificación, que pueden ser recuperados después por un dispositivo de decodificación en cualquier momento tras haberse almacenado en este medio.

[0038] Las normas de codificación de vídeo incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluidas sus ampliaciones de Codificación de Vídeo Escalable (SVC) y de Codificación de Vídeo Multivista (MVC). Además, la Codificación de Vídeo de Alta Eficacia (HEVC) se ha desarrollado recientemente por el Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC. Un borrador de la especificación HEVC está disponible en: http://phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zip, más adelante en el presente documento "JCTVC-N1003."

[0039] En HEVC y otras especificaciones de codificación de vídeo, una secuencia de vídeo incluye típicamente una serie de imágenes. Las imágenes también se pueden denominar "cuadros". Una imagen puede incluir tres matrices de muestras, denominadas S_L , S_{Cb} y S_{Cr} . S_L es una matriz bidimensional (es decir, un bloque) de muestras de luma. S_{Cb} es una matriz bidimensional de muestras de crominancia Cb. S_{Cr} es una matriz bidimensional de muestras de crominancia Cr. Las muestras de crominancia también se pueden denominar en el presente documento muestras de "croma". En otros casos, una imagen puede ser monocromática y puede incluir solo una matriz de muestras de luma.

[0040] Para generar una representación codificada de una imagen, el codificador de vídeo 20 puede generar un conjunto de unidades de árbol de codificación (CTU). Cada una de las CTU puede comprender un bloque de árbol de codificación de muestras de luma, dos bloques de árbol de codificación correspondientes de muestras de croma y estructuras sintácticas usadas para codificar las muestras de los bloques de árbol de codificación. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una CTU puede comprender un solo bloque de árbol de codificación y estructuras sintácticas usadas para codificar las muestras del bloque de árbol de codificación. Un bloque de árbol de codificación puede ser un bloque $N \times N$ de muestras. Una CTU también puede denominarse "bloque de árbol" o "unidad de codificación de máximo tamaño" (LCU). Las CTU de HEVC pueden ser análogas, en términos generales, a los macrobloques de otras normas, tales como la H.264/AVC. Sin embargo, una CTU no está necesariamente limitada a un tamaño en particular y puede incluir una o más unidades de codificación (CU). Un fragmento puede incluir un número entero de CTU ordenadas consecutivamente en un orden de exploración de barrido. Por ejemplo, un fragmento puede ser un número entero de CTU contenidas en un segmento de fragmento independiente y todos los segmentos de fragmento dependientes subsiguientes (si los hubiera) que preceden al siguiente segmento de fragmento independiente (si lo hubiera) dentro de la misma unidad de acceso.

[0041] Para generar una CTU codificada, el codificador de vídeo 20 puede realizar de forma recursiva una división de árbol cuaternario en los bloques de árbol de codificación de una CTU para dividir los bloques de árbol de codificación en bloques de codificación, de ahí el nombre "unidades de árbol de codificación". Un bloque de

codificación es un bloque NxN de muestras. Una CU puede comprender un bloque de codificación de muestras de luma y dos bloques de codificación correspondientes de muestras de croma de una imagen que tiene una matriz de muestras de luma, una matriz de muestras de Cb y una matriz de muestras de Cr y estructuras sintácticas usadas para codificar las muestras de los bloques de codificación. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una CU puede comprender un único bloque de codificación y estructuras sintácticas usadas para codificar las muestras del bloque de codificación.

[0042] El codificador de vídeo 20 puede dividir un bloque de codificación de una CU en uno o más bloques de predicción. Un bloque de predicción es un bloque rectangular (es decir, cuadrado o no cuadrado) de muestras en las que se aplica la misma predicción. Una unidad de predicción (PU) de una CU puede comprender un bloque de predicción de muestras de luma, dos bloques de predicción correspondientes de muestras de croma y estructuras sintácticas usadas para predecir los bloques de predicción. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una PU puede comprender un único bloque de predicción y estructuras sintácticas usadas para predecir el bloque de predicción. El codificador de vídeo 20 puede generar bloques predictivos de luma, Cb y Cr para bloques de predicción de luma, Cb y Cr de cada PU de la CU.

[0043] El codificador de vídeo 20 puede usar intrapredicción o interpredicción para generar los bloques predictivos para una PU. Si el codificador de vídeo 20 usa intrapredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras descodificadas de la imagen asociada a la PU.

[0044] Si el codificador de vídeo 20 usa interpredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras descodificadas de una o más imágenes distintas a la imagen asociada a la PU. La interpredicción puede ser una interpredicción unidireccional (es decir, unipredicción) o una interpredicción bidireccional (es decir, bipredicción). Para realizar la unipredicción o la bipredicción, el codificador de vídeo 20 puede generar una primera lista de imágenes de referencia (RefPicList0) y una segunda lista de imágenes de referencia (RefPicList1) para un fragmento actual. Cada una de las listas de imágenes de referencia puede incluir una o más imágenes de referencia. Cuando se usa unipredicción, el codificador de vídeo 20 puede buscar las imágenes de referencia en RefPicList0 y/o RefPicList1 para determinar una ubicación de referencia dentro de una imagen de referencia. Además, cuando se usa unipredicción, el codificador de vídeo 20 puede generar, basándose, al menos en parte, en muestras correspondientes a la ubicación de referencia, los bloques predictivos de muestras para la PU. Además, cuando se usa la unipredicción, el codificador de vídeo 20 puede generar un único vector de movimiento que indique un desplazamiento espacial entre un bloque de predicción de la PU y la ubicación de referencia. Para indicar el desplazamiento espacial entre un bloque de predicción de la PU y la ubicación de referencia, un vector de movimiento puede incluir un componente horizontal que especifique un desplazamiento horizontal entre el bloque de predicción de la PU y la ubicación de referencia y puede incluir un componente vertical que especifique un desplazamiento vertical entre el bloque de predicción de la PU y la ubicación de referencia.

[0045] Cuando se usa bipredicción para codificar una PU, el codificador de vídeo 20 puede determinar una primera ubicación de referencia en una imagen de referencia en RefPicList0 y una segunda ubicación de referencia en una imagen de referencia en RefPicList1. El codificador de vídeo 20 puede generar entonces, basándose, al menos en parte, en muestras correspondientes a las primera y segunda ubicaciones de referencia, los bloques predictivos para la PU. Además, cuando se usa la bipredicción para codificar la PU, el codificador de vídeo 20 puede generar un primer vector de movimiento que indica un desplazamiento espacial entre un bloque de muestra de la PU y la primera ubicación de referencia y un segundo vector de movimiento que indica un desplazamiento espacial entre el bloque de predicción de la PU y la segunda ubicación de referencia.

[0046] Típicamente, una estructura de lista de imágenes de referencia para la primera o la segunda lista de imágenes de referencia (por ejemplo, RefPicList0 o RefPicList1) de una imagen B incluye dos etapas: inicialización de lista de imágenes de referencia y reordenamiento (modificación) de la lista de imágenes de referencia. La inicialización de lista de imágenes de referencia es un mecanismo explícito que introduce las imágenes de referencia en la memoria de imágenes de referencia (también conocida como memoria intermedia de imágenes descodificadas) como una lista en base al orden de los valores POC (recuento de orden de imágenes, alineado con el orden de visualización de una imagen). El mecanismo de reordenación de lista de imágenes de referencia puede modificar la posición de una imagen que se introdujo en la lista durante la inicialización de lista de imágenes de referencia en cualquier posición nueva, o introducir cualquier imagen de referencia en la memoria de imágenes de referencia en cualquier posición, incluso si la imagen no pertenece a la lista inicializada. Algunas imágenes, después del reordenamiento (modificación) de la lista de imágenes de referencia, pueden colocarse en posiciones muy lejanas en la lista. Sin embargo, si la posición de una imagen supera un número de imágenes de referencia activas de la lista, la imagen no se considera una entrada de la lista final de imágenes de referencia. El número de imágenes de referencia activas se puede señalar en la cabecera de fragmento para cada lista. Después de que se construya una lista de imágenes de referencia (concretamente, RefPicList0 y RefPicList1, si procede), se puede usar un índice de referencia a una lista de imágenes de referencia para identificar cualquier imagen de referencia incluida en la lista de imágenes de referencia.

[0047] El codificador de vídeo 20 puede codificar determinados bloques de datos de vídeo usando codificación en modo de intrapredicción y proporcionar información que indica un modo de intrapredicción seleccionado usado para

codificar el bloque. El codificador de vídeo 20 puede codificar mediante intrapredicción bloques de cualquier tipo de cuadro o fragmento (por ejemplo, cuadros I o fragmentos I, además de cuadros P o fragmentos P y cuadros B o fragmentos B) usando un modo de intrapredicción. Cuando el codificador de vídeo 20 determina que un bloque debería codificarse en modo de intrapredicción, el codificador de vídeo 20 puede realizar un análisis de velocidad-distorsión para seleccionar el modo de intrapredicción más apropiado. Los modos de intrapredicción también pueden denominarse "intramodos". Por ejemplo, el codificador de vídeo 20 puede calcular valores de velocidad-distorsión para uno o más modos de intrapredicción y seleccionar uno de los modos que tenga características de velocidad-distorsión aceptables.

[0048] El codificador de vídeo 20 puede, en algunos ejemplos, estar configurado para comenzar el análisis para la selección de un modo de intrapredicción con el modo más probable, basándose en el contexto. Cuando el modo más probable alcanza características de velocidad-distorsión adecuadas, en algunos ejemplos, el codificador de vídeo 20 puede seleccionar el modo más probable. En otros ejemplos, el codificador de vídeo 20 no necesita comenzar el proceso de selección con el modo más probable.

[0049] Después de que el codificador de vídeo 20 genere un bloque predictivo (por ejemplo, un bloque predictivo de luma, Cb y Cr) para una o más PU de una CU, el codificador de vídeo 20 puede generar un bloque residual de luma para la CU. Cada muestra en un bloque residual puede indicar una diferencia entre una muestra en un bloque predictivo y una muestra correspondiente en un bloque de codificación original. Cada muestra en el bloque residual de luma de la CU indica una diferencia entre una muestra de luma en un bloque de los bloques de luma predictivos de la PU de la CU y una muestra correspondiente en el bloque de codificación de luma original de la CU. Además, el codificador de vídeo 20 puede generar un bloque residual de Cb para la CU. Cada muestra del bloque residual de Cb de la CU puede indicar una diferencia entre una muestra de Cb en uno de los bloques predictivos de Cb de la CU y una muestra correspondiente del bloque de codificación de Cb original de la CU. El codificador de vídeo 20 también puede generar un bloque residual de Cr para la CU. Cada muestra del bloque residual de Cr de la CU puede indicar una diferencia entre una muestra de Cr de uno de los bloques predictivos de Cr de la CU y una muestra correspondiente del bloque de codificación de Cr original de la CU.

[0050] Además, el codificador de vídeo 20 puede usar una división de árbol cuaternario para descomponer los bloques residuales de luma, Cb y Cr de una CU en uno o más bloques de transformada de luma, Cb y Cr. Un bloque de transformada es un bloque rectangular (por ejemplo, cuadrado o no cuadrado) de muestras a las que se aplica la misma transformada. Una unidad de transformada (TU) de una CU puede comprender un bloque de transformada de muestras de luma, dos bloques de transformada correspondientes de muestras de croma y estructuras sintácticas usadas para transformar las muestras de bloques de transformada. Por tanto, cada TU de una CU puede estar asociada a un bloque de transformada de luma, un bloque de transformada de Cb y un bloque de transformada de Cr. El bloque de transformada de luma asociado a la TU puede ser un subbloque del bloque residual de luma de la CU. El bloque de transformada de Cb puede ser un subbloque del bloque residual de Cb de la CU. El bloque de transformada de Cr puede ser un subbloque del bloque residual de Cr de la CU. En imágenes monocromáticas o imágenes que tienen tres planos de color separados, una TU puede comprender un único bloque de transformada y estructuras sintácticas usadas para transformar las muestras del bloque de transformada.

[0051] El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de luma de una TU para generar un bloque de coeficientes de luma para la TU. Un bloque de coeficientes puede ser una matriz bidimensional de coeficientes de transformada. Un coeficiente de transformada puede ser una cantidad escalar. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cb de una TU para generar un bloque de coeficientes de Cb para la TU. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cr de una TU para generar un bloque de coeficientes de Cr para la TU.

[0052] Después de generar un bloque de coeficientes (por ejemplo, un bloque de coeficientes de luma, un bloque de coeficientes de Cb o un bloque de coeficientes de Cr), el codificador de vídeo 20 puede cuantificar el bloque de coeficientes. La cuantificación se refiere, en general, a un proceso en el que coeficientes de transformada se cuantifican para reducir, posiblemente, la cantidad de datos usados para representar los coeficientes de transformada, proporcionando más compresión.

[0053] Por tanto, tras una codificación intrapredictiva o interpredictiva para generar datos predictivos y datos residuales, y tras cualquier transformada (tal como la transformada de enteros 4x4 u 8x8 usada en la norma H.264/AVC o una transformada discreta del coseno DCT) para generar coeficientes de transformada, se puede realizar cuantificación de los coeficientes de transformada. La cuantificación se refiere, en general, a un proceso en el que los coeficientes de transformada se cuantifican para reducir, posiblemente, la cantidad de datos usados para representar los coeficientes. El proceso de cuantificación puede reducir la profundidad de bits asociada con algunos, o la totalidad, de los coeficientes. Por ejemplo, un valor de n bits puede redondearse a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m .

[0054] Después de que el codificador de vídeo 20 cuantifique un bloque de coeficientes, el codificador de vídeo 20 puede codificar por entropía elementos sintácticos que indican los coeficientes de transformada cuantificados. Por ejemplo, el codificador de vídeo 20 puede realizar una codificación aritmética binaria adaptativa al contexto (CABAC)

en los elementos sintácticos que indican los coeficientes de transformada cuantificados. Por ejemplo, tras la cuantificación, puede realizarse la codificación por entropía de los datos cuantificados, por ejemplo, de acuerdo con la codificación de longitud variable adaptativa al contenido (CAVLC), CABAC u otra metodología de codificación por entropía. Una unidad de procesamiento configurada para la codificación por entropía, u otra unidad de procesamiento, puede desempeñar otras funciones de procesamiento, tales como la codificación de longitud de series de ceros de coeficientes cuantificados y/o la generación de información sintáctica, tal como valores de patrón de bloque codificado (CBP), tipo de macrobloque, modo de codificación, tamaño máximo de macrobloque para una unidad codificada (tal como un cuadro, un fragmento, un macrobloque o una secuencia) o similares.

[0055] El codificador de vídeo 20 también puede estar configurado para determinar un contexto de codificación para el bloque. El contexto puede determinarse en base a diversas características del bloque tales como, por ejemplo, el tamaño del bloque, que pueden determinarse en cuanto a dimensiones de píxel, tipo de unidad de predicción (PU) tal como, en el ejemplo de HEVC, $2N \times 2N$, $N \times 2N$, $2N \times N$, $N \times N$, tipos de intrapredicción de corta distancia (SDIP) tales como $2N \times N/2$, $N/2 \times 2N$, $2N \times 1$, $1 \times 2N$, un tipo de macrobloque en el ejemplo de H.264, una profundidad de unidad de codificación (CU) para el bloque, u otras mediciones de tamaño para un bloque de datos de vídeo. En algunos ejemplos, el codificador de vídeo 20 puede determinar el contexto basándose en algunos o todos los modos de intrapredicción para un bloque vecino superior, un bloque vecino izquierdo, un bloque vecino superior izquierdo, un bloque vecino superior derecho u otros bloques vecinos. En algunos ejemplos, el codificador de vídeo 20 determina el contexto basándose tanto en modos de intrapredicción para uno o más bloques como en información de tamaño para el bloque actual que se está codificando.

[0056] El codificador de vídeo 20 puede proporcionar un flujo de bits que incluye una secuencia de bits que forma una representación de imágenes codificadas y datos asociados. El flujo de bits puede comprender una secuencia de unidades de capa de abstracción de red (NAL). Una unidad NAL es una estructura sintáctica que contiene una indicación del tipo de datos en la unidad NAL y octetos que contienen esos datos en forma de una carga útil de secuencia de octetos sin procesar (RBSP) entremezclados como sea necesario con bits de prevención de emulación. Cada una de las unidades NAL incluye una cabecera de unidad NAL y encapsula una RBSP. La cabecera de unidad NAL puede incluir un elemento sintáctico que indica un código de tipo de unidad NAL. El código de tipo de unidad NAL especificado por la cabecera de unidad NAL de una unidad NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contiene un número entero de octetos que están encapsulados dentro de una unidad NAL. En algunos casos, una RBSP incluye bits cero.

[0057] Diferentes tipos de unidades NAL pueden encapsular diferentes tipos de RBSP. Por ejemplo, un primer tipo de unidad NAL puede encapsular una RBSP para un conjunto de parámetros de imagen (PPS), un segundo tipo de unidad NAL puede encapsular una RBSP para un fragmento codificado, un tercer tipo de unidad NAL puede encapsular una RBSP para mensajes SEI, etc. Las unidades NAL que encapsulan las RBSP para datos de codificación de vídeo (a diferencia de las RBSP para conjuntos de parámetros y mensajes SEI) se pueden denominar unidades NAL de capa de codificación de vídeo (VCL). Las unidades NAL que contienen conjuntos de parámetros (por ejemplo, conjuntos de parámetros de vídeo (VPS), conjuntos de parámetros de secuencia (SPS), conjuntos de parámetros de imagen (PPS), etc.) pueden denominarse unidades NAL de conjuntos de parámetros.

[0058] El descodificador de vídeo 30 puede recibir un flujo de bits generado por el codificador de vídeo 20. Además, el descodificador de vídeo 30 puede analizar sintácticamente el flujo de bits para obtener elementos sintácticos a partir del flujo de bits. El descodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo en base a, al menos en parte, los elementos sintácticos obtenidos a partir del flujo de bits. El proceso para reconstruir los datos de vídeo puede ser, en general, recíproco al proceso realizado por el codificador de vídeo 20. Por ejemplo, el descodificador de vídeo 30 puede usar vectores de movimiento de las PU para determinar bloques predictivos para las PU de una CU actual. Además, el descodificador de vídeo 30 puede cuantificar de forma inversa bloques de coeficientes asociados a las TU de la CU actual. El descodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes para reconstruir bloques de transformada asociados a las TU de la CU actual. El descodificador de vídeo 30 puede reconstruir los bloques de codificación de la CU actual añadiendo las muestras de los bloques predictivos para las PU de la CU actual a muestras correspondientes de los bloques de transformada de las TU de la CU actual. Reconstruyendo los bloques de codificación para cada CU de una imagen, el descodificador de vídeo 30 puede reconstruir la imagen.

[0059] Por lo tanto, en algunos ejemplos de esta divulgación, el descodificador de vídeo 30 puede, en última instancia, recibir datos de vídeo codificados, por ejemplo, desde la interfaz de entrada 28. De acuerdo con algunas técnicas de esta divulgación, el descodificador de vídeo 30 puede recibir una palabra de código u otra sintaxis representativa de un modo de intrapredicción usado para codificar un bloque de datos de vídeo. El descodificador de vídeo 30 puede estar configurado para determinar un contexto de codificación para el bloque de una manera sustancialmente similar al codificador de vídeo 20.

[0060] Como se mencionó anteriormente, las unidades NAL particulares del flujo de bits pueden incluir VPS, SPS y PPS. En algunos ejemplos, un VPS es una estructura sintáctica que comprende elementos sintácticos que se aplican a cero o más secuencias de vídeo codificadas (CVS) completas. En algunos ejemplos, un SPS es una estructura sintáctica que contiene elementos sintácticos que se aplican a cero o más CVS completas. Un SPS puede incluir un

elemento sintáctico que identifica un VPS que está activo cuando el SPS está activo. Por tanto, los elementos sintácticos de un VPS pueden ser más aplicables, en general, que los elementos sintácticos de un SPS. En algunos ejemplos, un PPS es una estructura sintáctica que contiene elementos sintácticos que se aplican a cero o más imágenes codificadas completas según lo determina un elemento sintáctico que se encuentra en cada cabecera de segmento de fragmento.

[0061] Un conjunto de parámetros (por ejemplo, un VPS, SPS, PPS, etc.) puede contener una identificación a la que se hace referencia, de forma directa o indirecta, desde una cabecera de fragmento de un fragmento. En algunos ejemplos, una cabecera de fragmento es la cabecera de segmento de fragmento del segmento de fragmento independiente que es un segmento de fragmento actual o el segmento de fragmento independiente más reciente que precede a un segmento de fragmento dependiente actual en orden de descodificación. En dichos ejemplos, una cabecera de segmento de fragmento es un número entero de unidades de árbol de codificación ordenadas consecutivamente en una exploración en mosaico y contenidas en una única unidad NAL. El proceso de referencia se conoce como "activación". Por tanto, cuando el descodificador de vídeo 30 está descodificando un fragmento particular, se dice que un conjunto de parámetros, al que un elemento sintáctico de una cabecera de fragmento del fragmento particular hace referencia de forma directa o indirecta, se "activa". Dependiendo del tipo de conjunto de parámetros, la activación puede producirse por imagen o por secuencia. Por ejemplo, una cabecera de fragmento de un fragmento puede incluir un elemento sintáctico que identifica un PPS. Por tanto, cuando un codificador de vídeo codifica el fragmento, el PPS se puede activar. Además, el PPS puede incluir un elemento sintáctico que identifica un SPS. Por tanto, cuando se activa un PPS que identifica el SPS, el SPS se puede activar. El SPS puede incluir un elemento sintáctico que identifica un VPS. Por tanto, cuando se activa un SPS que identifica el VPS, el VPS se activa.

[0062] Como se mencionó antes brevemente, el codificador de vídeo 20 puede codificar elementos sintácticos usando la codificación CABAC. Para aplicar la codificación CABAC a un elemento sintáctico, el codificador de vídeo 20 puede binarizar el elemento sintáctico para formar una serie de uno o más bits, que se denominan "*bins*". Además, el codificador de vídeo 20 puede identificar un contexto de codificación. El contexto de codificación puede identificar probabilidades de codificación de *bins* que tengan valores particulares. Por ejemplo, un contexto de codificación puede indicar una probabilidad de 0,7 de codificar un *bin* de valor 0 y una probabilidad de 0,3 de codificar un *bin* de valor 1. Después de identificar el contexto de codificación, el codificador de vídeo 20 puede dividir un intervalo en un subintervalo inferior y en un subintervalo superior. Uno de los subintervalos puede estar asociado al valor 0 y el otro subintervalo puede estar asociado al valor 1. El ancho de los subintervalos puede ser proporcional a las probabilidades indicadas para los valores asociados por el contexto de codificación identificado. Si un *bin* del elemento sintáctico tiene el valor asociado al subintervalo inferior, el valor codificado puede ser igual al límite inferior del subintervalo inferior. Si el mismo *bin* del elemento sintáctico tiene el valor asociado al subintervalo superior, el valor codificado puede ser igual al límite inferior del subintervalo superior. Para codificar el siguiente *bin* del elemento sintáctico, el codificador de vídeo 20 puede repetir estas etapas, donde el intervalo es el subintervalo asociado al valor del bit codificado. Cuando el codificador de vídeo 20 repite estas etapas para el siguiente *bin*, el codificador de vídeo 20 puede usar probabilidades modificadas en base a las probabilidades indicadas por el contexto de codificación identificado y los valores reales de los *bins* codificados. En algunos ejemplos, cuando el codificador de vídeo 20 repite estas etapas para el siguiente *bin*, el codificador de vídeo 20 puede seleccionar otro contexto de codificación.

[0063] Cuando el descodificador de vídeo 30 realiza la descodificación CABAC en un elemento sintáctico, el descodificador de vídeo 30 puede identificar un contexto de codificación. El descodificador de vídeo 30 puede dividir entonces un intervalo en un subintervalo inferior y un subintervalo superior. Uno de los subintervalos puede estar asociado al valor 0 y el otro subintervalo puede estar asociado al valor 1. El ancho de los subintervalos puede ser proporcional a las probabilidades indicadas para los valores asociados por el contexto de codificación identificado. Si el valor codificado está dentro del subintervalo inferior, el descodificador de vídeo 30 puede descodificar un *bin* que tenga el valor asociado al subintervalo inferior. Si el valor codificado está dentro del subintervalo superior, el descodificador de vídeo 30 puede descodificar un *bin* que tenga el valor asociado con el subintervalo superior. Para descodificar el siguiente *bin* del elemento sintáctico, el descodificador de vídeo 30 puede repetir estas etapas, donde el intervalo es el subintervalo que contiene el valor codificado. Cuando el descodificador de vídeo 30 repite estas etapas para el siguiente *bin*, el descodificador de vídeo 30 puede usar probabilidades modificadas en base a las probabilidades indicadas por el contexto de codificación identificado y los *bins* descodificados. El descodificador de vídeo 30 puede desbinarizar entonces los *bins* para recuperar el elemento sintáctico.

[0064] En lugar de llevar a cabo una codificación CABAC convencional en todos los elementos sintácticos, el codificador de vídeo 20 puede codificar algunos *bins* usando codificación CABAC por derivación. Puede ser computacionalmente menos costoso realizar una codificación CABAC por derivación en un *bin* que realizar una codificación CABAC convencional en el *bin*. Además, llevar a cabo la codificación CABAC por derivación puede permitir un mayor grado de paralelización y rendimiento. Los *bins* codificados usando codificación CABAC por derivación pueden denominarse "*bins* de derivación". Agrupar *bins* de derivación juntos puede aumentar el rendimiento del codificador de vídeo 20 y del descodificador de vídeo 30. El motor de codificación CABAC por derivación puede codificar varios *bins* en un solo ciclo, mientras que el motor de codificación CABAC convencional puede codificar solo un único *bin* en un ciclo. El motor de codificación CABAC por derivación puede ser más simple porque el motor de codificación CABAC por derivación no selecciona contextos y puede suponer una probabilidad de $\frac{1}{2}$ para ambos

símbolos (0 y 1). En consecuencia, en la codificación CABAC por derivación, los intervalos se dividen directamente a la mitad.

[0065] La FIG. 2 es un diagrama conceptual que ilustra un ejemplo de intrapredicción de un bloque de datos de vídeo. En la intrapredicción, un bloque de datos de vídeo (por ejemplo, una PU) se predice usando muestras de imágenes reconstruidas que son vecinas espacialmente. En la FIG. 2 se muestra un ejemplo típico de la intrapredicción para un bloque de imagen 40, por ejemplo, un bloque de imagen de 16x16. Con intrapredicción, el bloque de imagen 40 se predice por las muestras reconstruidas vecinas superior e izquierda (muestras de referencia) a lo largo de una dirección de predicción seleccionada (como se indica mediante la flecha 42).

[0066] La FIG. 3 es un diagrama conceptual que ilustra un ejemplo de modos de intrapredicción e índices de modo correspondientes. En HEVC hay 35 modos para la intrapredicción de un bloque de luma, incluido un modo planar (es decir, un modo de intrapredicción planar), un modo DC y 33 modos angulares (es decir, modos de intrapredicción angulares), como se indica en la FIG 3. Los 35 modos de la intrapredicción, tal como se define en HEVC, se indexan como se muestra en la siguiente tabla

Tabla 1 - Especificación del modo de intrapredicción y nombres asociados

Modo de intrapredicción	Nombre asociado
0	INTRA_PLANAR
1	INTRA_DC
2..34	INTRA_ANGULAR2..INTRA_ANGULAR34

[0067] La FIG. 4 es un diagrama conceptual que ilustra una técnica de ejemplo para generar una muestra de predicción para un bloque de datos de vídeo de acuerdo con un modo de intrapredicción planar. El modo planar es típicamente el modo de intrapredicción usado con más frecuencia. Para realizar una predicción planar en un bloque NxN, para cada muestra p_{xy} ubicada en (x, y), como se ilustra en la FIG. 4, el valor de predicción se calcula usando cuatro muestras reconstruidas vecinas específicas, es decir, muestras de referencia, con un filtro bilineal. Las cuatro muestras de referencia incluyen una muestra reconstruida superior derecha TR (50), una muestra reconstruida inferior izquierda BL (52) y las dos muestras reconstruidas (54, 56) ubicadas en la misma columna ($r_{x,-1}$) y fila ($r_{-1,y}$) de la muestra actual, como se ilustra en la FIG. 4. El modo planar se puede formular como: $p_{xy} = (N-x-1) \cdot L + (N-y-1) \cdot T + x \cdot TR + y \cdot BL$. En esta fórmula, N es la altura y el ancho del bloque.

[0068] En un modo DC, el bloque de predicción simplemente se llena con el valor promedio de las muestras reconstruidas vecinas. Por ejemplo, para generar un bloque predictivo para una PU usando un modo de intrapredicción DC, el codificador de vídeo puede establecer cada muestra del bloque predictivo para la PU igual a un valor promedio de muestras reconstruidas vecinas. En general, los modos de intrapredicción planares y DC se aplican para modelar regiones de imagen constantes y que varían suavemente.

[0069] HEVC especifica 33 direcciones de predicción diferentes para sus modos de intrapredicción angulares. Para cada intrapredicción angular dada, la dirección de intrapredicción puede identificarse, por ejemplo, de acuerdo con la FIG. 3. En el ejemplo de la FIG. 3, el modo de intrapredicción 10 corresponde a una dirección de predicción horizontal pura, y el modo de intrapredicción 26 corresponde a una dirección de predicción vertical pura.

[0070] Dada una dirección de intrapredicción específica, para cada muestra del bloque de predicción, su coordenada (x, y) se proyecta primero a la fila/columna de muestras reconstruidas vecinas a lo largo de la dirección de predicción, como se muestra en un ejemplo en la FIG. 3. Supóngase que (x, y) se proyecta en la posición fraccionaria α entre dos muestras reconstruidas vecinas L y R, después se calcula el valor de predicción para (x, y) usando un filtro de interpolación bilineal de dos etapas, formulado de la siguiente manera: $p_{xy} = (1-\alpha) \cdot L + \alpha \cdot R$. Para evitar operaciones de coma flotante, en HEVC, el cálculo anterior se aproxima usando aritmética de enteros como: $p_{xy} = ((32-\alpha) \cdot L + \alpha \cdot R + 16) \gg 5$, donde α es un número entero igual a $32 \cdot \alpha$.

[0071] En la codificación intramodo en HEVC, para cada intra-PU, se señala un modo de intrapredicción seleccionado. Para señalar el modo de intrapredicción seleccionado, primero se identifican tres modos de intrapredicción que se supone que tienen mayor probabilidad de seleccionarse realmente, a saber, los modos más probables (MPM). En HEVC, los MPM, etiquetados como `candModeList[x]`, $x=0, 1, 2$, se obtienen de la siguiente manera.

[0072] En primer lugar, un modo vecino izquierdo (`candIntraPredModeA`) y un modo vecino superior (`candIntraPredModeB`) se obtienen de la siguiente manera de acuerdo con la subcláusula 8.4.2 de JCTVC-N1003: La entrada para este proceso es una ubicación de luma (x_{Pb} , y_{Pb}) que especifica la muestra superior izquierda del bloque de predicción de luma actual en relación con la muestra de luma superior izquierda de la imagen actual.

1. Las ubicaciones vecinas (x_{NbA} , y_{NbA}) y (x_{NbB} , y_{NbB}) se establecen iguales a ($x_{Pb} - 1$, y_{Pb}) y (x_{Pb} , $y_{Pb} - 1$), respectivamente.

2. Para que X sea reemplazada por A o B, las variables candIntraPredModeX se obtienen de la siguiente manera:

- El proceso de obtención de disponibilidad para un bloque en orden de exploración en z como se especifica en la subcláusula 6.4.1 se invoca con la ubicación (xCurr, yCurr) establecida igual a (xPb, yPb) y la ubicación vecina (xNbY, yNbY) establecida igual a (xNbX, yNbX) como entradas, y la salida se asigna a availableX.
- El modo de intrapredicción candidato candIntraPredModeX se obtiene de la siguiente manera:
 - Si availableX es igual a FALSE, candIntraPredModeX se establece igual a INTRA_DC.
 - De lo contrario, si CuPredMode[xNbX][yNbX] no es igual a MODE_INTRA o pcm_flag[xNbX][yNbX] es igual a 1, candIntraPredModeX se establece igual a INTRA_DC,
 - De lo contrario, si X es igual a B e yPb - 1 es menor que ((yPb >> CtbLog2SizeY) << CtbLog2SizeY), candIntraPredModeB se establece igual a INTRA_DC.
 - De lo contrario, candIntraPredModeX se establece igual a IntraPredModeY[xNbX][yNbX].

[0073] Por lo tanto, si la muestra vecina a la izquierda de la muestra superior izquierda del bloque de predicción de luma de una PU actual no está disponible, si la muestra vecina a la izquierda de la muestra superior izquierda del bloque de predicción de luma de la PU actual no se predice usando intrapredicción, o si la muestra vecina a la izquierda de la muestra superior izquierda del bloque de predicción de luma de la PU actual se codifica usando modulación de código de pulso (PCM), candIntraPredModeA se establece en el modo de intrapredicción DC. De lo contrario, candIntraPredModeA se establece en el modo de intrapredicción de una PU cuyo bloque de predicción contiene la muestra vecina a la izquierda de la muestra superior izquierda del bloque de predicción de luma de la PU actual. Del mismo modo, si la muestra vecina por encima de la muestra superior izquierda del bloque de predicción de luma de una PU no está disponible, si la muestra vecina por encima de la muestra superior izquierda del bloque de predicción de luma de la PU actual no se predice usando intrapredicción, la muestra vecina por encima de la muestra superior izquierda del bloque de predicción de luma de la PU actual se codifica usando PCM, o si la muestra vecina por encima de la muestra superior izquierda del bloque de predicción de luma de la PU actual está en un bloque de árbol de codificación diferente de la PU actual, candIntraPredModeB se establece en el modo de intrapredicción DC. De lo contrario, candIntraPredModeB se establece en el modo de intrapredicción de luma de una PU cuyo bloque de predicción contiene la muestra vecina a la izquierda de la muestra superior izquierda del bloque de predicción de luma de la PU actual (es decir, IntraPredModeY[xNbX][yNbX]).

[0074] A continuación, en la subcláusula 8.4.2 de JCTVC-N1003, usando el modo vecino izquierdo obtenido (candIntraPredModeA) y el modo vecino superior (candIntraPredModeB), los tres MPM se obtienen de la siguiente manera:

3. candModeList[x] con x = 0..2 se obtiene como sigue:

- Si candIntraPredModeB es igual a candIntraPredModeA, se aplica lo siguiente:
 - Si candIntraPredModeA es menor que 2 (es decir, igual a INTRA_PLANAR o INTRA_DC), candModeList[x] con x = 0..2 se obtiene como sigue:

candModeList[0] = INTRA_PLANAR (8-15)

candModeList[1] = INTRA_DC (8-16)

candModeList[2] = INTRA_ANGULAR26 (8-17)
 - De lo contrario, candModeList[x] con x = 0..2 se obtiene como sigue:

candModeList[0] = candIntraPredModeA (8-18)

candModeList[1] = 2 + ((candIntraPredModeA + 29) % 32) (8-19)

candModeList[2] = 2 + ((candIntraPredModeA - 2 + 1) % 32) (8-20)
- De lo contrario (candIntraPredModeB no es igual a candIntraPredModeA), se aplica lo siguiente:
 - candModeList[0] y candModeList[1] se obtienen como sigue:

candModeList[0] = candIntraPredModeA (8-21)

candModeList[1] = candIntraPredModeB (8-22)

- Si ninguno de candModeList[0] y candModeList[1] es igual a INTRA_PLANAR, candModeList[2] se establece igual a INTRA_PLANAR,

- De lo contrario, si ninguno de candModeList[0] y candModeList[1] es igual a INTRA_DC, candModeList[2] se establece igual a INTRA_DC,

- De lo contrario, candModeList[2] se establece igual a INTRA_ANGULAR26.

[0075] Después de que se deciden los tres MPM, en la subcláusula 8.4.2 de JCTVC-N1003, para cada PU, se señala primero un indicador de un bit prev_intra_luma_pred_flag[xPb][yPb] para indicar si el intramodo seleccionado de la PU actual es el mismo que uno de los tres MPM.

- Si prev_intra_luma_pred_flag[xPb][yPb] se señala como 1, es decir, se selecciona uno de los 3 MPM para codificar la PU actual, un índice mpm_idx (puede ser igual a 0, 1 o 2), que indica qué MPM se seleccionó para codificar la PU actual, se señala adicionalmente. El mpm_idx se binariza usando código unario truncado y se codifica por derivación usando modelado sin contexto.

- De lo contrario (prev_intra_luma_pred_flag[xPb][yPb] se señala como 0), es decir, no se usa MPM para la PU actual, un índice rem_intra_luma_pred_mode[xPb][yPb], que indica el modo no MPM que está seleccionado para la PU actual, se señala adicionalmente. El valor de rem_intra_luma_pred_mode[xPb][yPb] podría ser 0, 1,..., 31, y la binarización de longitud fija (5 bits) se usa con la codificación por derivación.

[0076] Como se analizó previamente, el codificador de vídeo 20 selecciona el modo de intrapredicción para una PU actual. El proceso de selección del modo de intrapredicción para una PU puede denominarse decisión de intramodo en el lado del codificador. Cuando se selecciona un modo de intrapredicción para una PU actual, el codificador de vídeo 20 puede seleccionar uno de los tres MPM determinados para la PU actual o uno de los modos de intrapredicción no MPM. Si el codificador de vídeo 20 selecciona uno de los tres MPM para la PU actual como el modo de intrapredicción para la PU actual, el codificador de vídeo 20 puede señalar, en el flujo de bits, un índice MPM para la PU actual. En JCTVC-N1003, el índice MPM para una PU que tiene una muestra de luma superior izquierda en las coordenadas (x, y) se denota como mpm_ind[x][y]. De lo contrario, si el codificador de vídeo 20 no selecciona uno de los tres MPM para la PU actual como el modo de intrapredicción para la PU actual, el codificador de vídeo 20 puede señalar, en el flujo de bits, un índice no MPM para la PU actual. En JCTVC-N1003, el índice no MPM para una PU que tiene una muestra de luma superior izquierda en las coordenadas (x, y) se denota como rem_intra_pred_mode[x][y]. Además, el codificador de vídeo 20 señala, en el flujo de bits, un elemento sintáctico que indica si se señala un índice MPM o un índice no MPM para la PU actual. En JCTVC-N1003, este elemento sintáctico para una PU que tiene una muestra de luma superior izquierda en las coordenadas (x, y) se denota como prev_intra_luma_pred_flag[x][y].

[0077] En JCTVC-N1003, un índice MPM solo puede tener tres valores diferentes posibles, mientras que un índice no MPM puede tener muchos más valores potenciales. En consecuencia, se pueden requerir menos bits para señalar un índice MPM que un índice no MPM.

[0078] Para seleccionar un modo de intrapredicción para una PU actual de la manera más eficiente posible, en el diseño del software de referencia HEVC (denominado "HM"), se han integrado varios procedimientos rápidos de decisión de intramodo. El software de referencia HM se puede descargar desde: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-14.0/.

[0079] Dado que hay hasta 35 modos de intrapredicción disponibles para un bloque de luma, una decisión del modo de codificador basada en una optimización total de velocidad-distorsión puede ser demasiado costosa para una implementación práctica. En la última versión de HM, la decisión intramodo se realiza en dos fases. En la primera fase, una lista de *N* candidatos a intramodo se decide primero, de manera aproximada, usando un criterio de coste mucho más económico conocido comúnmente como "suma de diferencias de transformada absolutas" (SATD). El valor de *N* depende del tamaño del bloque, y la configuración por defecto en HM es: *N* es igual a 8 para 4x4 y 8x8, y *N* es igual a 3 para bloques de 16x16 y más grandes. Después, los modos de vecino izquierdo y superior, es decir, candIntraPredModeA o tanto candIntraPredModeA como candIntraPredModeB (si candIntraPredModeA no es igual a candIntraPredModeA), se añaden a la lista de candidatos a intramodo, si no están ya incluidos. En la segunda fase, la lista de candidatos a intramodo se introduce en el costoso proceso de cálculo del coste de velocidad-distorsión, y el mejor modo final de intrapredicción se decide para la PU actual. Con este proceso de decisión intramodo de dos fases, la mayoría de modos de intrapredicción se omiten para el costoso cálculo del coste de velocidad-distorsión, y el mejor modo de intrapredicción se selecciona aún sin una penalización excesiva por la caída del rendimiento de codificación.

[0080] En el documento de Matsuo *et al.*, "Improved intra angular prediction by DCT-based interpolation filter", Signal Processing Conference (EUSIPCO), 2012 Actas de la vigésima conferencia europea, páginas 1568-1572. IEEE, 2012,

se propone aplicar un filtro de interpolación basado en DCT de 4 etapas para tamaños de bloque de 4x4 y 8x8, y el filtro de suavizado intra también se desactiva cuando se aplica el filtro de 4 etapas; para tamaños de bloque mayores o iguales a 16x16, se aplica el filtro de interpolación bilineal de 2 etapas. En el documento de Maani, Ehsan, "Interpolation filter for intra prediction of HEVC", solicitud de patente de EE. UU. 13/312.946, presentada el 6 de diciembre de 2011, se puede usar un filtro de interpolación de 4 etapas cuando el filtro de suavizado intra está desactivado, mientras que el filtro de interpolación de 4 etapas podría obtenerse en base a un proceso de interpolación CUBIC, un proceso de interpolación basado en DCT o un proceso de interpolación de Hermite. En el documento de M. Guo, X. Guo y S. Lei, "Improved Intra Mode Coding", Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, Cuarta conferencia: Daegu, Corea, del 20 al 28 de enero de 2011, la binarización del intramodo se selecciona de manera adaptativa a partir de un conjunto de árboles de codificación predefinidos de acuerdo con los modos de sus bloques vecinos.

[0081] Puede haber una serie de problemas asociados a las técnicas existentes de intrapredicción para codificar datos de vídeo, por ejemplo, de acuerdo con la norma HEVC. Por ejemplo, los 35 ángulos de predicción fijos del diseño existente de intrapredicción en HEVC pueden ser ineficaces para capturar distribuciones de dirección de borde muy flexibles. Sin embargo, la aplicación de más ángulos de predicción puede dar como resultado un aumento no deseado de la complejidad del codificador. Por ejemplo, una ampliación directa a 65 ángulos de predicción en base al software de referencia de HEVC puede requerir un número aproximadamente duplicado de verificaciones SATD con respecto a los 35 ángulos de predicción fijos del diseño existente de intrapredicción en HEVC. Además, la eficacia de la codificación intramodo actual, por ejemplo, usando tres MPM en HEVC, puede ser limitada ya que puede no corresponder exactamente con la distribución de probabilidad real de todos los modos de intrapredicción disponibles. En otras palabras, los tres MPM determinados en HEVC para una PU particular pueden no ser, en realidad, los modos de intrapredicción que probablemente se usarán para la PU particular.

[0082] Las técnicas de esta divulgación pueden resolver los problemas identificados anteriormente asociados a las técnicas existentes de intrapredicción para la codificación de datos de vídeo, por ejemplo, de acuerdo con la norma HEVC, así como otros problemas en los campos técnicos de la intrapredicción y la codificación y compresión de vídeo. Los siguientes procedimientos detallados pueden aplicarse individualmente. De forma alternativa, se puede aplicar cualquier combinación de los mismos.

[0083] De acuerdo con algunas técnicas de esta divulgación, se pueden usar más de tres MPM para señalar un modo de intrapredicción. Por ejemplo, un ejemplo típico del número de MPM, denotado por M , puede ser 4, 5 o 6. El uso de MPM adicionales puede reducir la probabilidad de usar índices no MPM para señalar modos de intrapredicción de PU. Debido a que los índices MPM se representan típicamente usando menos bits que los índices no MPM, el uso de MPM adicionales puede aumentar la disminución del tamaño de flujo de bits.

[0084] En algunos ejemplos en los que se usan más de tres MPM para señalar un modo de intrapredicción de una PU, el codificador de vídeo 20 puede usar un proceso de codificación por entropía, tal como CABAC, para codificar un índice MPM y el decodificador de vídeo 30 puede usar un proceso de decodificación por entropía, tal como CABAC, para decodificar el índice MPM. El índice MPM para una PU puede ser un valor numérico que identifica cuál de las MPM es el modo de intrapredicción de la PU. Como parte del uso del proceso de codificación por entropía para codificar el índice MPM, el codificador de vídeo 20 puede binarizar el índice MPM. En otras palabras, el codificador de vídeo 20 puede convertir el índice MPM en un código binario. En diferentes ejemplos, el codificador de vídeo 20 puede usar diversos procedimientos de binarización para convertir el índice MPM en un código binario. Por ejemplo, un codificador de vídeo puede binarizar el índice MPM usando al menos uno de: un código unario truncado fijo o un código Huffman fijo. Además, en algunos ejemplos, cuando se codifica el índice MPM, el procedimiento de binarización puede ser un código unario truncado fijo, un código Huffman fijo o se decide en base a los intramodos vecinos codificados. Por ejemplo, un codificador de vídeo puede determinar el procedimiento de binarización basándose en modos de intrapredicción usados para codificar uno o más bloques vecinos. Por ejemplo, para un índice MPM de una PU, el codificador de vídeo puede determinar el procedimiento de binarización basándose en modos de intrapredicción usados para codificar uno o más bloques vecinos contiguos a la PU. Después de binarizar el índice MPM, el codificador de vídeo 20 puede codificar mediante CABAC el índice MPM binarizado e incluir el índice MPM binarizado codificado por CABAC en el flujo de bits. Como parte de la decodificación del índice MPM, el decodificador de vídeo 30 puede obtener el índice MPM binarizado codificado por CABAC a partir del flujo de bits, aplicar la decodificación CABAC al índice MPM binarizado codificado por CABAC para recuperar el índice MPM binarizado y desbinarizar el índice MPM para recuperar el índice MPM. En esta divulgación, la expresión "bins de un índice MPM" puede referirse a bins del índice MPM binarizado.

[0085] En algunos ejemplos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar modelado de contexto cuando codifican (por ejemplo, codifican mediante CABAC) uno o más bins de un índice MPM. En otras palabras, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden seleccionar, para cada bin respectivo de uno o más bins del índice MPM, un contexto de codificación respectivo para el bin respectivo. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar el contexto de codificación respectivo para el bin respectivo para codificar el bin respectivo. Por ejemplo, cuando se codifica el índice MPM, los N primeros bins codificados o los N bins codificados seleccionados pueden codificarse usando modelado de contexto, mientras que el índice de contexto se decide usando los modos de intrapredicción vecinos codificados. Por ejemplo, para un índice

MPM de una PU en una imagen actual, el codificador de vídeo puede determinar el índice de contexto en base a los modos de intrapredicción usados para codificar una o más PU vecinas contiguas en la imagen actual a la PU. En este ejemplo, N es un número entero. En diversos ejemplos, N puede ser igual a 1, 2 o 3. Por lo tanto, en algunos ejemplos, el uno o más *bins* modelados por contexto consisten en el primer *bin* del índice MPM. En otros ejemplos, el uno o más de los uno o más *bins* modelados por contexto consisten en N *bins* seleccionados del índice MPM, donde N se establece igual a $M - K$, donde K es un número predefinido.

[0086] En un ejemplo en el que el codificador de vídeo 20 y el descodificador de vídeo 30 usan modelado de contexto al codificar uno o más *bins* de un índice MPM de una PU actual, el índice de contexto depende de los casos de los modos de intrapredicción vecinos izquierdo y superior. En el caso 0 de este ejemplo, los modos de intrapredicción vecinos izquierdo y superior son iguales y ninguno de los modos de intrapredicción es DC o planar. En el caso 1 de este ejemplo, los modos de intrapredicción vecinos izquierdo y superior son iguales y al menos uno de los modos de intrapredicción es DC o planar. En el caso 2 de este ejemplo, los modos de intrapredicción vecinos izquierdo y superior son diferentes y ninguno es planar. En el caso 3 de este ejemplo, los modos de intrapredicción vecinos izquierdo y superior son diferentes y al menos uno de los modos de intrapredicción es planar. Los modos de intrapredicción vecinos izquierdo y superior pueden ser los modos de intrapredicción de las PU izquierda y superior de la PU actual. En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden calcular el índice de contexto en base a una tabla de correlación predefinida que correlaciona desde el índice de caso (por ejemplo, caso 0, caso 1, caso 2 o caso 3) a un número de índice de contexto. En otras palabras, un codificador de vídeo puede usar una tabla de correlación de índice de caso a índice de contexto para determinar un índice de contexto a partir del índice de caso. El número de índice de contexto puede identificar un contexto de codificación predefinido en un conjunto de contextos de codificación predefinidos.

[0087] Por lo tanto, en este ejemplo, cuando un codificador de vídeo selecciona un índice de contexto para un *bin* modelado por contexto del índice MPM, el codificador de vídeo puede seleccionar un primer índice de contexto si los modos de intrapredicción usados para descodificar un bloque vecino izquierdo y un bloque vecino superior son los mismos y ninguno de los modos de intrapredicción usados para descodificar ni el bloque vecino izquierdo ni el bloque vecino superior es DC o planar. Además, en este ejemplo, el codificador de vídeo puede seleccionar un segundo índice de contexto si los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior son iguales y al menos uno de los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior es DC o planar. En este ejemplo, el codificador de vídeo puede seleccionar un tercer índice de contexto si los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior son diferentes y ninguno de los modos de intrapredicción usados para descodificar ni el bloque vecino izquierdo ni el bloque vecino superior es planar. En este ejemplo, en este ejemplo, el codificador de vídeo puede seleccionar un cuarto índice de contexto si los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior son diferentes y al menos uno de los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior es DC o planar.

[0088] En algunos ejemplos, se usan diferentes tablas de correlación de índice de caso a índice de contexto para diferentes *bins*. Por ejemplo, un codificador de vídeo, tal como el codificador de vídeo 20 o el descodificador de vídeo 30, puede usar una primera tabla de correlación de índice de caso a índice de contexto para el *bin* 0 del índice MPM, una segunda tabla de correlación de índice de caso a índice de contexto para el *bin* 1 del índice MPM, una tercera tabla de correlación de índice de caso a índice de contexto para el *bin* 2 del índice MPM, y así sucesivamente. Las diferentes tablas de correlación de índice de caso a índice de contexto de este ejemplo pueden asignar diferentes contextos a los casos (por ejemplo, caso 0, caso 1, caso 2 y caso 3) descritos anteriormente. Por ejemplo, la correlación descrita en el párrafo anterior puede aplicarse para el primer *bin* del índice MPM, pero para un segundo *bin* del índice MPM, el codificador de vídeo puede seleccionar el segundo índice de contexto si los modos de intrapredicción usados para descodificar un bloque vecino izquierdo y un bloque vecino superior son iguales y ninguno de los modos de intrapredicción usados para descodificar el bloque vecino izquierdo o el bloque vecino superior es DC o planar; seleccionar el tercer índice de contexto si los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior son iguales y al menos uno de los modos de intrapredicción usados para descodificar el bloque vecino izquierdo y el bloque vecino superior es DC o planar; y así sucesivamente.

[0089] En algunos ejemplos en los que el codificador de vídeo 20 y el descodificador de vídeo 30 usan modelado de contexto al codificar uno o más *bins* de un índice MPM de una PU actual, solo se aplica un contexto para codificar cada uno de los N primeros *bins* o N *bins* seleccionados, y el contexto puede no depender de los modos de intrapredicción vecinos. Por ejemplo, en dichos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 realizan el proceso de selección de contexto de codificación una vez para el índice MPM y no repiten el proceso de selección de contexto de codificación en los siguientes *bins* del índice MPM. En otras palabras, un codificador de vídeo puede seleccionar un índice de contexto común para cada uno de los uno o más *bins* modelados por contexto del índice MPM.

[0090] Como se mencionó anteriormente, en algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar modelado de contexto al codificar N *bins* codificados de un índice MPM. En uno de dichos ejemplos, N se establece igual a $(M-K)$, donde K es un número predefinido y M es el número de MPM. Por ejemplo, K puede ser igual a 1, 2 o 3.

[0091] Por lo tanto, en los ejemplos anteriores, el codificador de vídeo 20 puede obtener M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En este ejemplo, M es mayor que 3. Además, se codifica un elemento sintáctico que indica si se usa un índice de modo más probable o un índice de modo que no es más probable para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en que el índice indicado de entre el índice MPM o el índice no MPM sea el índice MPM, el codificador de vídeo 20 puede seleccionar, para cada *bin* del uno o más *bins* modelados por contexto del índice MPM, en base a modos de intrapredicción usados para descodificar uno o más bloques vecinos, un índice de contexto para el *bin* modelado por contexto. El codificador de vídeo 20 puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0092] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, para codificar el elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar el modo de intrapredicción seleccionado, el codificador de vídeo 20 puede codificar un elemento sintáctico (por ejemplo, *prev_intra_luma_pred_flag*) que indica si un índice MPM (por ejemplo, *mpm_idx*) o un índice no MPM (por ejemplo, *rem_intra_luma_pred_mode*) se usa para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción de la PU. El codificador de vídeo 20 puede, para cada *bin* de los uno o más *bins* modelados por contexto del índice MPM, usar el índice de contexto seleccionado para el *bin* modelado por contexto para realizar la codificación CABAC del *bin* modelado por contexto. Además, como parte de la codificación del bloque de datos de vídeo basada en el modo de intrapredicción seleccionado, el codificador de vídeo 20 puede generar, basándose en el modo de intrapredicción seleccionado de la PU actual, un bloque predictivo para la PU actual. El codificador de vídeo 20 puede generar datos residuales que representan diferencias de píxel entre la CU actual y el bloque predictivo. En este ejemplo, el codificador de vídeo 20 puede proporcionar un flujo de bits que incluye una secuencia de bits que forma una representación de imágenes codificadas y datos asociados, donde las imágenes codificadas incluyen la imagen actual.

[0093] De forma similar, en algunos ejemplos, el codificador de vídeo 30 puede obtener M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En este ejemplo, M es mayor que 3. Además, el descodificador de vídeo 30 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado y el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en que el índice indicado de entre el índice MPM o el índice no MPM sea el índice MPM, el descodificador de vídeo 30 puede seleccionar, para cada *bin* del uno o más *bins* modelados por contexto del índice MPM, en base a modos de intrapredicción usados para descodificar uno o más bloques vecinos, un índice de contexto para el *bin* modelado por contexto. Además, el descodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0094] Por ejemplo, en este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, para descodificar el elemento sintáctico que indica si el índice MPM o el índice no MPM se usa para indicar el modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede descodificar, a partir de un flujo de bits, un elemento sintáctico (por ejemplo, *prev_intra_luma_pred_flag*) que indica si se usa un índice MPM (por ejemplo, *mpm_idx*) o un índice no MPM (por ejemplo, *rem_intra_luma_pred_mode*) para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción de la PU actual. En este ejemplo, para cada *bin* de los uno o más *bins* modelados por contexto del índice MPM, el descodificador de vídeo 30 puede usar el índice de contexto seleccionado para el *bin* modelado por contexto para realizar la descodificación CABAC del *bin* modelado por contexto del índice MPM. Además, como parte de la reconstrucción del bloque de datos de vídeo en base al modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede generar un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado. En este ejemplo, el descodificador de vídeo 30 puede reconstruir la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual. En este ejemplo, el descodificador de vídeo 30 puede recibir un flujo de bits que incluye una secuencia de bits que forma una representación de imágenes codificadas y datos asociados, donde las imágenes codificadas incluyen la imagen actual.

[0095] Como se analizó antes anteriormente, la eficacia de la codificación en modo de intrapredicción usando los tres MPM especificados en HEVC puede ser limitada ya que, con frecuencia, los tres MPM especificados en HEVC no corresponden con exactitud a la distribución de probabilidad real de todos los modos de intrapredicción disponibles. Por ejemplo, la probabilidad de que el modo de intrapredicción de una PU sea un modo de intrapredicción particular que no sea MPM puede ser mayor que las probabilidades de que el modo de intrapredicción de la PU sea cualquiera de los tres MPM especificados en HEVC.

[0096] Por lo tanto, de acuerdo con algunos ejemplos de esta divulgación, cuando se obtienen los MPM, se define un modo de intrapredicción representativo para la columna (o fila) vecina izquierda (o superior) y el modo de intrapredicción representativo se usa como el candidato a MPM desde la columna vecina izquierda (o superior). Por ejemplo, un codificador de vídeo puede definir un modo de intrapredicción representativo para una columna vecina izquierda y usar el modo representativo para la columna vecina izquierda como candidato a MPM. El codificador de vídeo puede usar el modo representativo para la columna vecina izquierda como candidato a MPM además del MPM definido en HEVC para el bloque vecino izquierdo o en lugar del MPM definido en HEVC para el bloque vecino izquierdo (candIntraPredModeA). Además, el codificador de vídeo puede definir un modo de intrapredicción representativo para una fila vecina superior y usar el modo representativo para la fila vecina superior como candidato a MPM. El codificador de vídeo puede usar el modo representativo para la fila vecina superior como candidato a MPM además del MPM definido en HEVC para el bloque vecino superior o en lugar del MPM definido en HEVC para el bloque vecino superior (candIntraPredModeB).

[0097] La columna vecina izquierda puede incluir una columna de muestras inmediatamente a la izquierda en una imagen de una columna de muestras que contiene una muestra actual de una PU actual para una CU de la imagen. Por ejemplo, la columna vecina izquierda puede incluir (o consistir en) cada muestra inmediatamente a la izquierda de la PU actual. La fila vecina superior puede incluir una fila de muestras inmediatamente superiores en una imagen de una fila de muestras que contiene una muestra actual de una PU actual para una CU de la imagen. Por ejemplo, la fila vecina superior puede incluir (o consistir en) cada muestra inmediatamente superior a la PU actual.

[0098] Por ejemplo, como parte de la codificación de datos de vídeo, el codificador de vídeo 20 puede codificar un bloque de datos de vídeo en parte obteniendo M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En algunos ejemplos, M es mayor que 3. En este ejemplo, los MPM pueden incluir un MPM para la columna vecina izquierda y un MPM para la fila vecina superior. En algunos ejemplos, el codificador de vídeo 20 define un modo de intrapredicción representativo para la columna vecina izquierda y usa el modo de intrapredicción representativo para la columna vecina izquierda como MPM para la columna vecina izquierda. En algunos ejemplos, el codificador de vídeo 20 define un modo de intrapredicción representativo para la fila vecina superior y usa el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior.

[0099] Además, el codificador de vídeo 20 puede codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo, y codifica el índice indicado del índice MPM o el índice no MPM. Por ejemplo, el codificador de vídeo 20 puede incluir, en el flujo de bits, datos que indican un elemento sintáctico `prev_intra_luma_pred_flag` que indica si el flujo de bits incluye datos que indican un elemento sintáctico `mpm_index` o `rem_intra_luma_pred_mode`. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de los modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. El codificador de vídeo 20 codifica el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0100] En este ejemplo, el bloque de datos de vídeo es una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. Para codificar el bloque de datos de vídeo en base al modo de intrapredicción seleccionado, el codificador de vídeo 20 puede reconstruir el bloque de datos de vídeo, en parte, generando un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado. Además, el codificador de vídeo 20 puede reconstruir la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual.

[0101] En un ejemplo similar de selección de un modo de intrapredicción a usar en la intrapredicción para codificar datos de vídeo, el descodificador de vídeo 30 puede descodificar un bloque de datos de vídeo, en parte, obteniendo M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. En este ejemplo, M puede ser mayor que 3. Los MPM pueden incluir un MPM para la columna vecina izquierda y un MPM para la fila vecina superior. En algunos ejemplos, el descodificador de vídeo 30 define un modo de intrapredicción representativo para la columna vecina izquierda y usa el modo de intrapredicción representativo para la columna vecina izquierda como MPM para la columna vecina izquierda. En algunos ejemplos, el descodificador de vídeo 30 define un modo de intrapredicción representativo para la fila vecina superior y usa el modo de intrapredicción representativo para la fila vecina superior como MPM para la fila vecina superior.

[0102] Además, el descodificador de vídeo 30 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo, y descodifica el índice indicado del índice de modo más probable o el índice de modo que no es más probable. Por ejemplo, el descodificador de vídeo 30 puede obtener un `prev_intra_pred_flag` a partir de un flujo de bits. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado.

[0103] El descodificador de vídeo 30 puede reconstruir entonces un bloque de vídeo basándose en el modo de intrapredicción seleccionado resultante. En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, como parte de la reconstrucción del bloque de vídeo basada en el modo de intrapredicción resultante seleccionado, el descodificador de vídeo 30 puede usar el modo de intrapredicción seleccionado para generar un bloque predictivo para el bloque de vídeo. Por ejemplo, el bloque de vídeo puede ser una CU y el descodificador de vídeo 30 puede usar el modo de intrapredicción seleccionado para generar un bloque predictivo para una PU de la CU. Además, el descodificador de vídeo 30 puede reconstruir al menos algunas muestras del bloque de vídeo añadiendo muestras del bloque predictivo a las muestras residuales correspondientes para el bloque de vídeo. Por ejemplo, el bloque de vídeo puede ser una CU y el descodificador de vídeo 30 puede añadir muestras del bloque predictivo a muestras residuales correspondientes en uno o más bloques de transformada de las TU de la CU.

[0104] En diversos ejemplos, un codificador de vídeo puede determinar un modo de intrapredicción representativo de diferentes maneras. En un ejemplo, se define en función de todos los modos de intrapredicción usados por los bloques vecinos. Por ejemplo, se cuenta el número de modos de intrapredicción realmente usados de la columna (o fila) vecina izquierda (o superior), en unidad del tamaño de PU más pequeño (por ejemplo, 4), y el usado con más frecuencia se define como el modo de intrapredicción representativo. Por lo tanto, en los ejemplos anteriores, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden definir el modo de intrapredicción representativo para la columna vecina izquierda basándose en el modo de intrapredicción más frecuente de la columna vecina izquierda, y definir el modo de intrapredicción representativo para la columna vecina superior basándose en el modo de intrapredicción más frecuente de la fila vecina superior.

[0105] De forma alternativa, el recuento se basa en el tamaño de PU del bloque actual. Por ejemplo, un codificador de vídeo puede determinar el modo de intrapredicción más frecuente de los bloques vecinos basándose en al menos uno de: (i) el tamaño de unidad de predicción más pequeño, o (ii) un tamaño de unidad de predicción del bloque de datos de vídeo. En un ejemplo, en lugar de contar en base a la unidad del tamaño de PU más pequeño (por ejemplo, 4x4), se puede usar un tamaño de unidad diferente para un tamaño de PU diferente del bloque actual. Por ejemplo, si el tamaño de PU del bloque actual es 64x64, el recuento se basa en un tamaño de PU de 8x8 o un tamaño de PU de 16x16, en lugar del tamaño de PU más pequeño (4x4).

[0106] En algunos ejemplos, un modo de intrapredicción representativo puede definirse como uno de los modos de intrapredicción usados por un bloque vecino seleccionado en la columna (o fila) vecina izquierda (o superior). Por ejemplo, el modo de intrapredicción de la muestra (o bloque igual al tamaño de PU más pequeño) ubicada en la posición media de la columna (fila) vecina izquierda (derecha), que no es la situada más a la izquierda (más arriba) o más a la derecha (más abajo), se define como el modo de intrapredicción representativo del modo vecino izquierdo (o superior). Además, en los ejemplos anteriores, el modo de intrapredicción más frecuente de los bloques vecinos puede determinarse en base a, al menos, uno del tamaño de unidad de predicción más pequeño o un tamaño de unidad de predicción del bloque de datos de vídeo. En otro ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden definir el modo de intrapredicción representativo para la columna vecina izquierda como un modo de intrapredicción usado por un bloque vecino seleccionado en la columna vecina izquierda, y pueden definir el modo de intrapredicción representativo para la fila vecina superior como un modo de intrapredicción usado por un bloque vecino seleccionado en la fila vecina superior.

[0107] Como se mencionó anteriormente, ejemplos particulares de esta divulgación permiten M MPM, donde M es mayor que tres. Por lo tanto, de acuerdo con dichos ejemplos de esta divulgación, el codificador de vídeo 20 puede obtener M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En otras palabras, el codificador de vídeo 20 puede obtener, de entre la pluralidad de modos de intrapredicción, M MPM para la intrapredicción de una PU de una CU de una imagen actual de los datos de vídeo. En este ejemplo, M es mayor que 3. En este ejemplo, M es menor que un número total de modos de intrapredicción en la pluralidad de modos de intrapredicción. Además, en este ejemplo, el codificador de vídeo 20 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. En este ejemplo, el codificador de vídeo 20 puede codificar el índice indicado entre el índice MPM o el índice no MPM. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado y el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. El codificador de vídeo 20 puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0108] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, el codificador de vídeo 20 puede codificar el elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar el modo de intrapredicción seleccionado, el codificador de vídeo 20 puede codificar un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica si un índice MPM (por ejemplo, `mpm_idx`) o un índice no MPM (por ejemplo, `rem_intra_luma_pred_mode`) se usa para indicar un modo de intrapredicción seleccionado de la pluralidad de modos

de intrapredicción para la intrapredicción de la PU. Además, como parte de la codificación del bloque de datos de vídeo basada en el modo de intrapredicción seleccionado, el codificador de vídeo 20 puede generar, basándose en el modo de intrapredicción seleccionado de la PU actual, un bloque predictivo para la PU actual. El codificador de vídeo 20 puede generar datos residuales que representan diferencias de píxel entre la CU actual y el bloque predictivo. En este ejemplo, el codificador de vídeo 20 puede proporcionar un flujo de bits que incluye una secuencia de bits que forma una representación de imágenes codificadas y datos asociados, donde las imágenes codificadas incluyen la imagen actual.

[0109] Además, de acuerdo con dichos ejemplos de esta divulgación, el descodificador de vídeo 30 puede obtener M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En este ejemplo, M es mayor que 3. M puede ser menor que el número total de modos de intrapredicción en la pluralidad de modos de intrapredicción. Además, en este ejemplo, el descodificador de vídeo 30 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. En este ejemplo, el descodificador de vídeo 30 puede descodificar el índice indicado entre el índice MPM o el índice no MPM. El descodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0110] Por ejemplo, en este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, para descodificar el elemento sintáctico que indica si el índice MPM o el índice no MPM se usa para indicar el modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede descodificar, a partir de un flujo de bits, un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica si se usa un índice MPM (por ejemplo, `mpm_idx`) o un índice no MPM (por ejemplo, `rem_intra_luma_pred_mode`) para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción de la PU actual. Además, como parte de la reconstrucción del bloque de datos de vídeo en base al modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede generar un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado. En este ejemplo, el descodificador de vídeo 30 puede reconstruir la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual. En este ejemplo, el descodificador de vídeo 30 puede recibir un flujo de bits que incluye una secuencia de bits que forma una representación de imágenes codificadas y datos asociados, donde las imágenes codificadas incluyen la imagen actual.

[0111] En algunos ejemplos, al decidir los M MPM ($M > 3$), los tres MPM originales definidos en HEVC se mantienen en los M MPM, y se añaden ($M-3$) modos de intrapredicción adicionales. Por ejemplo, en algunos ejemplos, los tres modos de intrapredicción adicionales se pueden seleccionar de modo que los modos planar y DC siempre se incluyan en los MPM. En otras palabras, el proceso de obtención de los M MPM puede comprender incluir un modo planar y un modo DC entre los M MPM.

[0112] De forma alternativa, en algunos ejemplos, los ($M-3$) modos de intrapredicción adicionales se pueden seleccionar de modo que los modos planar, DC, VERTICAL (es decir, INTRA_ANGULAR26) y/o HORIZONTAL (es decir, INTRA_ANGULAR10), y/o DIAGONAL (es decir, INTRA_ANGULAR18) siempre se incluyan en los MPM. En otras palabras, el proceso de obtención de los M MPM puede comprender incluir un modo Planar, un modo DC y al menos uno de un modo VERTICAL, un modo HORIZONTAL o un modo DIAGONAL entre los M MPM.

[0113] En algunos ejemplos, los ($M-3$) modos de intrapredicción adicionales se pueden seleccionar entre los modos angulares más aproximados al modo angular ya incluido en los tres primeros MPM. En este ejemplo, el término "más aproximado" se mide por la diferencia de los índices de modo de intrapredicción o el valor absoluto de la diferencia de los índices de modo de intrapredicción, o por la diferencia de los ángulos de intrapredicción. Por ejemplo, en este ejemplo, como parte de obtener los M MPM, un codificador de vídeo (por ejemplo, el codificador de vídeo 20 o el descodificador de vídeo 30) puede obtener tres MPM, donde los tres MPM incluyen al menos un modo angular. Además, en este ejemplo, el codificador de vídeo puede seleccionar uno o más MPM angulares adicionales en base a la similitud con el al menos modo angular. En este ejemplo, la similitud se determina en base a al menos una de diferencias de índice de modo de intrapredicción o diferencias de ángulo de intrapredicción.

[0114] De acuerdo con algunos ejemplos de esta divulgación, para reducir el número de bits usados para codificar valores de índice, se pueden incluir determinadas variaciones en el número de bits que se utilizan para codificar valores de índice al codificar el índice no MPM. Por lo tanto, un codificador de vídeo, tal como el codificador de vídeo 20 y el descodificador de vídeo 30, puede determinar si se está usando un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de una pluralidad de modos de intrapredicción y, cuando se identifica un índice no MPM, usar un código de longitud fija para representar el índice no MPM o un código de longitud variable para representar el índice no MPM, dependiendo de si se cumplen uno o más criterios. En diversos ejemplos, un codificador de vídeo puede determinar los diversos criterios de diferentes maneras. Por ejemplo, los criterios pueden cumplirse si el índice no MPM es uno de un número predeterminado de primeros índices no MPM de la pluralidad de modos de intrapredicción. En algunos ejemplos, los criterios se cumplen si el índice no MPM es uno de un número

predeterminado de últimos índices no MPM. En algunos ejemplos, los criterios se cumplen si el índice no MPM es uno de un número predeterminado de índices no MPM seleccionados.

[0115] Por lo tanto, de acuerdo con un ejemplo de esta divulgación, el codificador de vídeo 20 puede obtener M modos más probables para la intrapredicción de un bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. En este ejemplo, M puede ser mayor que 3. Además, el codificador de vídeo 20 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. Además, en este ejemplo, el codificador de vídeo 20 puede codificar, basándose en el elemento sintáctico que indica que el índice no MPM se usa para indicar el modo de intrapredicción seleccionado, el índice no MPM. En este ejemplo, el índice no MPM puede codificarse en el flujo de bits como una palabra de código más corta que $\lceil \log_2 N \rceil$ bits si el índice no MPM satisface un criterio y, de lo contrario, se codifica como un código de longitud fija con $\lceil \log_2 N \rceil$ bits. En este ejemplo, puede haber un total de N valores disponibles del índice no MPM. Además, el codificador de vídeo 20 puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado. En esta divulgación, $\lceil \cdot \rceil$ representa la operación de redondeo por exceso.

[0116] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo puede ser un modo de intrapredicción seleccionado para una PU actual de la CU actual. Además, en este ejemplo, para codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar el modo de intrapredicción seleccionado, el codificador de vídeo 20 puede incluir, en un flujo de bits, datos que indican un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica si se usa un índice MPM (por ejemplo, `mpm_idx`) o un índice no MPM (por ejemplo, `rem_intra_luma_pred_mode`) para indicar, para la PU actual, el modo de intrapredicción seleccionado de una pluralidad de modos de intrapredicción. Además, en algunos de dichos ejemplos, para codificar el bloque de datos de vídeo en base al modo de intrapredicción seleccionado, el codificador de vídeo 20 puede generar un bloque predictivo para la PU usando el modo de intrapredicción seleccionado, y el codificador de vídeo 20 puede generar datos residuales que representan diferencias de píxeles entre la CU actual y el bloque predictivo. En algunos ejemplos, el codificador de vídeo 20 puede transformar y cuantificar los datos residuales e incluir, en un flujo de bits, elementos sintácticos codificados por entropía que representan los coeficientes de transformada cuantificados resultantes.

[0117] En un ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de primeros índices no MPM de la pluralidad de modos de intrapredicción. Por ejemplo, para modos no MPM, tanto el codificador como el descodificador pueden asignar un valor de índice a cada uno de los modos no MPM disponibles. Con el valor de índice descodificado, el descodificador sabrá qué modo no MPM está señalizado. En un ejemplo, cuando un índice no MPM es uno de los M primeros valores de índice, el índice no MPM puede estar dentro del intervalo de $0 \sim (M-1)$. En otro ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de últimos índices no MPM. En otro ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de índices no MPM seleccionados.

[0118] En un ejemplo similar, el codificador de vídeo 30 puede obtener M MPM para la intrapredicción del bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En este ejemplo, M puede ser mayor que 3. Además, en este ejemplo, el descodificador de vídeo 30 puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. Como antes, el índice MPM indica cuál de los M modos más probables es el modo de intrapredicción seleccionado y el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Además, en este ejemplo, en base al índice MPM que indica el modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede descodificar el índice no MPM. En este ejemplo, el índice no MPM se codifica en el flujo de bits como una palabra de código más corta que $\lceil \log_2 N \rceil$ bits si el índice no MPM cumple un criterio y se codifica en el flujo de bits como un código de longitud fija con un índice $\lceil \log_2 N \rceil$. El descodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0119] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo puede ser un modo de intrapredicción seleccionado para una PU actual de la CU actual. Además, en este ejemplo, para descodificar el elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar el modo de intrapredicción seleccionado para la intrapredicción de la PU actual, el descodificador de vídeo 30 puede descodificar u obtener de otro modo, a partir de un flujo de bits, un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica si se usa un índice MPM (por ejemplo, `mpm_idx`) o un índice no MPM (por ejemplo, `rem_intra_luma_pred_mode`) para indicar un modo de intrapredicción seleccionado para la PU actual. Además, en algunos ejemplos, como parte de la reconstrucción del bloque de datos de vídeo en base al modo de intrapredicción seleccionado, el descodificador de vídeo 30 puede generar un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado. En dichos ejemplos, el descodificador de vídeo 30 puede reconstruir la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual.

[0120] En un ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de primeros índices no MPM de la pluralidad de modos de intrapredicción. Por ejemplo, el criterio puede ser que el índice no MPM sea uno de los X primeros índices no MPM de la pluralidad de modos de intrapredicción, siendo X un número entero. En otro ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de últimos índices no MPM. Por ejemplo, el criterio puede ser que el índice no MPM sea uno de los X últimos índices no MPM de la pluralidad de modos de intrapredicción, siendo X un número entero. En otro ejemplo, el criterio puede cumplirse si el índice no MPM es uno de un número predeterminado de índices no MPM seleccionados. Por ejemplo, el criterio puede ser que el índice no MPM sea uno de los X índices no MPM seleccionados de la pluralidad de modos de intrapredicción, siendo X un número entero.

[0121] Por lo tanto, al codificar el índice no MPM, todos los N índices no MPM disponibles, excepto los M primeros o los M últimos o los M valores de índice seleccionados pueden codificarse usando un código de longitud fija con $\lceil \log_2 N \rceil$ bits, mientras que los M valores de índice se señalizan usando una palabra de código más corta que $\lceil \log_2 N \rceil$ bits. En este ejemplo, M puede ser igual, o no, al número de MPM. De forma alternativa, todos los N índices no MPM disponibles se codifican usando código de longitud fija con $\lceil \log_2 N \rceil$ bits.

[0122] De forma alternativa, en algunos ejemplos, un indicador de un bit, que indica si el índice no MPM se refiere a un modo horizontal o vertical, se señala primero con modelado de contexto y, a continuación, se señala adicionalmente un índice que especifica además el modo no MPM que se selecciona. Por lo tanto, en este ejemplo, el codificador de vídeo 20 puede codificar, basándose en un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica que el índice no MPM (por ejemplo, `rem_intra_luma_pred_mode`) se usa para indicar el modo de intrapredicción seleccionado para el bloque de datos de vídeo, un segundo elemento sintáctico que indica si el índice no MPM hace referencia a uno de un modo horizontal o un modo vertical. De manera similar, el decodificador de vídeo 30 puede decodificar, a partir del flujo de bits, en base a un elemento sintáctico (por ejemplo, `prev_intra_luma_pred_flag`) que indica que se usa un índice no MPM, un segundo elemento sintáctico que indica si el índice no MPM hace referencia a uno de un modo horizontal o un modo vertical.

[0123] En algunos ejemplos, cuando se usan más de tres MPM para señalar el modo de intrapredicción, durante la decisión del modo de codificador, para cada PU, los K ($K \geq 2$) primeros MPM se insertan en la lista de modos de intrapredicción candidatos que va a verificarse mediante optimización de velocidad-distorsión. El valor de K puede depender de si el modo de intrapredicción vecino izquierdo *IntraModeL* y el modo de intrapredicción vecino superior *IntraModeA* son los mismos. Por ejemplo, $K = (\text{IntraModeL} == \text{IntraModeA}) ? 2 : 3$. En otras palabras, en este ejemplo, K se establece igual a 2 si *IntraModeL* es igual a *IntraModeA* y, de lo contrario, se establece igual a 3.

[0124] En algunos ejemplos, el número de MPM puede depender de los tamaños de CU o PU. Por ejemplo, el número de MPM puede ser mayor para tamaños de CU o PU más grandes. De forma alternativa, en algunos ejemplos, el número de MPM puede señalizarse en un conjunto de parámetros de secuencia, un conjunto de parámetros de imagen o una cabecera de fragmento.

[0125] De acuerdo con algunos ejemplos de esta divulgación, un codificador de vídeo, tal como el codificador de vídeo 20 y el decodificador de vídeo 30, puede usar más de 33 modos de intrapredicción angulares. Por ejemplo, el codificador de vídeo puede usar 65 modos angulares. En este ejemplo, los 65 ángulos de predicción (direcciones) pueden definirse de modo que la interpolación se realice con una precisión de 1/32 pel (píxel). Como se analiza anteriormente, cuando una PU se codifica usando un modo de intrapredicción angular, un codificador de vídeo puede determinar una posición fraccionaria entre dos muestras vecinas reconstruidas proyectando, a lo largo de una dirección de predicción asociada al modo de intrapredicción angular, las coordenadas de una muestra del bloque predictivo de la PU. Cuando la interpolación se realiza con una precisión de 1/32 pel, se definen 31 posiciones equidistantes entre los dos píxeles vecinos reconstruidos contiguos y el codificador de vídeo establece la posición fraccionaria determinada como la más cercana de los píxeles vecinos reconstruidos o las 31 posiciones. El uso de un número limitado de posiciones en lugar de una línea de número real puede simplificar el procesamiento involucrado en la realización de los cálculos de interpolación. Como se describió antes brevemente, realizar una interpolación usando una precisión de 1/32 pel a pesar de que hay disponibles más de 33 modos de intrapredicción angulares, puede aumentar la precisión sin variar el nivel de complejidad. En particular, para realizar interpolaciones con mayor precisión, sería necesario diseñar y almacenar un mayor número de filtros de interpolación diferentes tanto en el codificador como en el decodificador. Por lo tanto, por ejemplo, al evitar la necesidad de una mayor precisión de interpolación, la complejidad (solo se utilizan 32 filtros de interpolación diferentes) del codificador de vídeo puede permanecer igual usando interpolaciones de 1/32 pel.

[0126] Por lo tanto, en los ejemplos anteriores donde se usan más de 33 modos de intrapredicción angulares (por ejemplo, se usan 65 módulos de intrapredicción angular) y la interpolación se realiza con una precisión de 1/32 pel, como parte de la codificación de un bloque de datos de vídeo, el codificador de vídeo 20 puede codificar información sintáctica (por ejemplo, `mpm_idx` o `rem_intra_luma_pred_mode`) que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo entre una pluralidad de modos de intrapredicción que incluyen más de 33 modos de intrapredicción angulares. En este ejemplo, los modos de intrapredicción angulares se definen de tal manera que la interpolación se realice con una precisión de 1/32 pel. Además, en este ejemplo, el codificador de vídeo 20 puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0127] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. Además, en este ejemplo, como parte de la codificación del bloque de datos de vídeo, el codificador de vídeo 20 puede, para cada muestra respectiva de un bloque predictivo de la PU actual, determinar una posición fraccionaria entre dos muestras reconstruidas vecinas de la imagen actual. La FIG. 5 es un diagrama conceptual que ilustra una técnica de ejemplo para generar una muestra de predicción para un bloque de datos de vídeo de acuerdo con un modo de intrapredicción angular. Por ejemplo, en el ejemplo de la FIG. 5, el codificador de vídeo 20 puede determinar una posición fraccionaria α entre las muestras vecinas L y R. Las dos muestras reconstruidas vecinas son vecinas de la PU actual. El codificador de vídeo 20 puede determinar la posición fraccionaria proyectando, a lo largo de una dirección de predicción asociada al modo de intrapredicción seleccionado, una coordenada de la muestra respectiva a una fila o columna de muestras reconstruidas vecinas que contienen las dos muestras reconstruidas vecinas. Por ejemplo, en el ejemplo de la FIG. 5, la línea 60 muestra la dirección de predicción asociada al modo de intrapredicción seleccionado y (x, y) es la coordenada de la muestra respectiva 62. En este ejemplo, el codificador de vídeo 20 puede calcular un valor de predicción de la muestra respectiva usando un filtro de interpolación que usa valores de las dos muestras reconstruidas vecinas para interpolar un valor en la posición fraccionaria determinada. Esta interpolación tiene una precisión de 1/32 pel. Además, en este ejemplo, como parte de la codificación del bloque de datos de vídeo, el codificador de vídeo 20 puede generar datos residuales que representan diferencias de píxeles entre el bloque actual y el bloque predictivo. El codificador de vídeo 20 puede transformar y cuantificar los datos residuales e incluir, en un flujo de bits, elementos sintácticos codificados por entropía que representan los coeficientes de transformada cuantificados resultantes.

[0128] De manera similar, en un ejemplo, el descodificador de vídeo 30 puede descodificar información sintáctica (por ejemplo, `mpm_idx` o `rem_intra_luma_pred_mode`) que indica un modo de intrapredicción seleccionado para un bloque de datos de vídeo entre una pluralidad de modos de intrapredicción. En este ejemplo, la pluralidad de modos de intrapredicción incluye más de 33 modos de intrapredicción angulares. Los modos de intrapredicción angulares pueden definirse de modo que la interpolación se realice con una precisión de 1/32 pel. El descodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado.

[0129] Por tanto, en este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo puede ser un modo de intrapredicción seleccionado para una PU actual de la CU actual. Como parte de la reconstrucción del bloque de datos de vídeo, para cada muestra respectiva de un bloque predictivo de la PU actual, el descodificador de vídeo 30 puede determinar una posición fraccionaria entre dos muestras reconstruidas vecinas de la imagen actual proyectando, a lo largo de una dirección de predicción asociada al modo de intrapredicción seleccionado, una coordenada de la muestra respectiva con respecto a una fila o columna de muestras reconstruidas vecinas que contienen las dos muestras reconstruidas vecinas. Además, el descodificador de vídeo 30 puede calcular un valor de predicción de la muestra respectiva usando un filtro de interpolación que usa valores de las dos muestras reconstruidas vecinas para interpolar un valor en la posición fraccionaria determinada. En este ejemplo, esta interpolación tiene una precisión de 1/32 pel. El descodificador de vídeo 30 puede reconstruir un bloque de codificación de la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual.

[0130] En los ejemplos anteriores, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar más de 33 modos de intrapredicción angulares. Por ejemplo, la pluralidad de modos de intrapredicción angulares puede incluir 65 modos de intrapredicción angulares. Además, en los ejemplos anteriores, el filtro de interpolación puede ser un filtro de interpolación bilineal de dos etapas. En algunos de dichos ejemplos, el filtro de interpolación se formula como:

$$p_{xy} = (1 - \alpha) \cdot L + \alpha \cdot R$$

donde p_{xy} es el valor calculado de la muestra respectiva, L y R son valores de las dos muestras vecinas reconstruidas, y α es la posición fraccionaria determinada.

[0131] Como se indicó anteriormente, de acuerdo con algunos ejemplos de esta divulgación, un codificador de vídeo puede usar más de 33 modos de intrapredicción angulares. En un ejemplo de este tipo, para cada PU, está restringido que solo se encuentren disponibles N modos angulares para su uso, que se seleccionan de forma adaptativa a partir de todos los modos angulares disponibles usando la información codificada de los bloques vecinos izquierdo y superior. Por ejemplo, en este ejemplo, el codificador de vídeo 20 tiene prohibido (es decir, restringido) seleccionar modos de intrapredicción angulares particulares cuando codifica un bloque (por ejemplo, PU) de una imagen. En este ejemplo, se puede prohibir que el codificador de vídeo 20 seleccione los modos de intrapredicción angulares particulares a pesar de que los modos de intrapredicción angulares particulares estén disponibles para que el codificador de vídeo 20 los use en la codificación de otros bloques de la misma imagen y a pesar del hecho de que los modos de intrapredicción angulares particulares puedan ofrecer un mejor rendimiento de compresión que los modos de intrapredicción angulares que el codificador de vídeo 20 puede usar para codificar el bloque.

[0132] Por lo tanto, en un ejemplo en el que solo están disponibles N modos de intrapredicción angulares para su uso, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden seleccionar, en base a los modos de intrapredicción usados para descodificar uno o más bloques vecinos, un subconjunto de N modos de intrapredicción angulares de entre la pluralidad de modos de intrapredicción angulares. En este ejemplo, la información sintáctica que indica el modo de intrapredicción seleccionado para la PU actual puede comprender un índice que indica el modo de intrapredicción seleccionado. En este ejemplo, el índice es un código de longitud fija que consiste en $\lceil \log_2 N \rceil$ bits. En este ejemplo, N puede ser diversos números enteros, tal como 33.

[0133] En algunos ejemplos, el número total de modos de intrapredicción depende de los tamaños de CU o PU. Por ejemplo, un codificador de vídeo (por ejemplo, el codificador de vídeo 20 o el descodificador de vídeo 30) puede determinar, basándose en al menos uno de un tamaño de CU o un tamaño de PU, un número total de modos de intrapredicción en la pluralidad de modos de intrapredicción. Por ejemplo, si un tamaño de CU o un tamaño de PU está por debajo de un umbral particular, solo están disponibles los modos de intrapredicción angulares con valores de índice inferiores a un número particular. En algunos ejemplos, el número total de modos de intrapredicción puede señalizarse en un SPS, un PPS o una cabecera de fragmento.

[0134] Como se analiza anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden aplicar un filtro de interpolación a muestras vecinas reconstruidas para determinar valores de muestras en un bloque predictivo. De acuerdo con algunas técnicas de esta divulgación, se puede aplicar un filtro de interpolación intra de N etapas, donde N es mayor que 2. Por ejemplo, N puede ser 3 y 4. Cuando un codificador de vídeo, tal como el codificador de vídeo 20 o el descodificador de vídeo 30, usa un filtro de interpolación intra de N etapas para determinar el valor de una muestra en un bloque predictivo, el codificador de vídeo puede calcular el valor basándose en los valores de N muestras vecinas reconstruidas. El uso de interpolación de N etapas ($N > 2$) puede ser ventajoso con respecto a la interpolación de 2 etapas existente, ya que puede proporcionar resultados de interpolación más exactos al incluir más muestras de referencia (N muestras de referencia) durante la interpolación. Estadísticamente, el uso de más muestras de referencia introducirá una mayor exactitud de interpolación con mayor complejidad. El filtro de interpolación intra de N etapas puede incluir un filtro de interpolación *sinc*, un filtro de interpolación gaussiano y un filtro de interpolación obtenido usando un modelo de correlación de imágenes. Un filtro *sinc* elimina todas las componentes de frecuencia por encima de una frecuencia de corte dada, sin afectar a las frecuencias más bajas, y tiene una respuesta de fase lineal. Por lo tanto, el filtro de interpolación de N niveles puede comprender al menos uno de un filtro de interpolación *sinc*, un filtro de interpolación gaussiano y un filtro de interpolación obtenido usando un modelo de correlación de imágenes.

[0135] De esta manera, el codificador de vídeo 20 puede codificar información sintáctica que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. Por ejemplo, el codificador de vídeo 20 puede señalizar, en un flujo de bits, un elemento sintáctico *mpm_idx* o *rem_intra_luma_pred_mode* para la PU actual. Además, el codificador de vídeo 20 puede aplicar un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas del bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado, donde N es mayor que 2. El codificador de vídeo 20 puede aplicar el filtro de interpolación intra de N etapas para determinar muestras reconstruidas vecinas filtradas que pueden corresponder a posiciones entre muestras reconstruidas vecinas de pel completo. El codificador de vídeo 20 puede codificar el bloque de datos de vídeo en base a las muestras reconstruidas vecinas filtradas de acuerdo con el modo de intrapredicción seleccionado. Por ejemplo, el codificador de vídeo 20 puede usar las muestras reconstruidas vecinas filtradas para generar un bloque predictivo y generar datos residuales para el bloque de datos de vídeo basándose en el bloque predictivo.

[0136] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. Además, en este ejemplo, para cada muestra respectiva de un bloque de predicción de la PU actual, el codificador de vídeo 20 puede determinar una posición fraccionaria entre dos muestras reconstruidas vecinas de un conjunto de muestras reconstruidas vecinas proyectando, a lo largo de una dirección de predicción asociada al modo de intrapredicción seleccionado, una coordenada de la muestra respectiva con respecto a una fila o columna de muestras reconstruidas vecinas que contienen las dos muestras reconstruidas vecinas. El conjunto de muestras reconstruidas vecinas incluye las muestras reconstruidas superior e izquierda de la PU actual en la imagen actual. Además, el codificador de vídeo 20 puede calcular un valor de la muestra respectiva aplicando un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas para interpolar un valor en la posición fraccionaria determinada, donde N es mayor que 2. Como parte de la codificación del bloque de datos de vídeo basada en las muestras reconstruidas vecinas filtradas, el codificador de vídeo 20 puede generar datos residuales que representan diferencias de píxeles entre la CU actual y el bloque predictivo. El codificador de vídeo 20 puede transformar y cuantificar los datos residuales e incluir, en un flujo de bits, elementos sintácticos codificados por entropía que representan los coeficientes de transformada cuantificados resultantes.

[0137] De forma similar, el descodificador de vídeo 30 puede descodificar información sintáctica que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. Por ejemplo, el descodificador de vídeo 30 puede obtener, a partir de un flujo de bits, un elemento sintáctico *mpm_idx* o *rem_intra_luma_pred_mode*. Además, el descodificador de vídeo 30 puede aplicar un filtro de interpolación intra de

N etapas a muestras reconstruidas vecinas del bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado, donde N es mayor que 2. Además, el decodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo en base a las muestras reconstruidas vecinas filtradas de acuerdo con el modo de intrapredicción seleccionado.

[0138] En este ejemplo, el bloque de datos de vídeo puede ser una CU actual en una imagen actual de los datos de vídeo y el modo de intrapredicción seleccionado para el bloque de datos de vídeo es un modo de intrapredicción seleccionado para una PU actual de la CU actual. En este ejemplo, para cada muestra respectiva de un bloque predictivo de la PU actual, el decodificador de vídeo 30 puede determinar una posición fraccionaria entre dos muestras reconstruidas vecinas de un conjunto de muestras reconstruidas vecinas proyectando, a lo largo de una dirección de predicción asociada al modo de intrapredicción seleccionado, una coordenada de la muestra respectiva con respecto a una fila o columna de muestras reconstruidas vecinas que contienen las dos muestras reconstruidas vecinas. El conjunto de muestras reconstruidas vecinas incluye las muestras reconstruidas superior e izquierda de la PU actual en la imagen actual. El decodificador de vídeo 30 puede calcular un valor de predicción de la muestra respectiva aplicando un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas para interpolar un valor en la posición fraccionaria determinada, donde N es mayor que 2. En este ejemplo, como parte de la reconstrucción del bloque de datos de vídeo (por ejemplo, la CU actual), el decodificador de vídeo 30 puede reconstruir un bloque de codificación de la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de las TU de la CU actual.

[0139] Cuando el filtro de interpolación se obtiene usando un modelo de correlación de imágenes, se puede aplicar una función de correlación de imágenes basada en la función gaussiana generalizada, y la función de interpolación para cada posición fraccionaria se obtiene adicionalmente usando la estimación cuadrática media mínima. Por ejemplo, cuando se obtiene el filtro de interpolación usando un modelo de correlación de imágenes, suponiendo que el modelo de correlación de imágenes es $R(i,j) = \rho_x^{|i|} \cdot \rho_y^{|j|}$, donde i y j indican la distancia entre dos muestras de imagen en el eje horizontal y el vertical, ρ_x es el factor de correlación de imágenes en dirección horizontal y ρ_y es el factor de correlación de imágenes en dirección vertical, tanto ρ_x como ρ_y pueden estar dentro del intervalo de $[0, 1]$, lo que significa que una distancia mayor entre imágenes genera una menor correlación. En otro ejemplo, se supone que hay cuatro muestras, a, b, c y d , y que la muestra fraccionaria e va a interpolarse. Para obtener un filtro de interpolación deseable $[f_0, f_1, f_2, f_3]$, el proceso de interpolación $[a, b, c, d] * [f_0, f_1, f_2, f_3]^T = [e]$ puede reescribirse como $A * F = E$, donde $A = [a, b, c, d]$, $F = [f_0, f_1, f_2, f_3]^T$, y $E = [e]$. De acuerdo con una estimación cuadrática media mínima, el filtro óptimo F es $F^* = (A^T A)^{-1} A^T E$, y $"(A^T A)^{-1} A^T E"$ puede determinarse mediante el modelo de correlación de imágenes supuesto. Por lo tanto, usando el modelo de correlación de imágenes, el filtro de interpolación para cada muestra fraccionaria se puede obtener usando una estimación cuadrática media mínima.

[0140] Por lo tanto, en algunos ejemplos en los que un codificador de vídeo obtiene el filtro de interpolación intra usando un modelo de correlación de imágenes, el codificador de vídeo puede aplicar una función de correlación de imágenes basada en la función gaussiana generalizada. Además, el codificador de vídeo puede obtener una función de interpolación para cada posición fraccionaria usando una estimación cuadrática media mínima.

[0141] En algunos ejemplos, cuando un codificador de vídeo aplica un filtro de interpolación gaussiano, el parámetro que controla la intensidad de suavizado del filtro puede variar dependiendo del tamaño de bloque. Por lo tanto, en dichos ejemplos, el filtro de interpolación intra puede comprender un filtro de interpolación gaussiano, y un parámetro que controla la intensidad de suavizado del filtro de interpolación gaussiano varía en base a los tamaños de bloque. Por ejemplo, el filtro de interpolación gaussiano puede formularse como:

$$G(x, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}},$$

donde

$G(x, \sigma)$, x indica la posición fraccionaria, y σ controla la intensidad de suavizado. Por lo tanto, un mayor valor de σ genera un filtro de interpolación gaussiano con una mayor intensidad de suavizado.

[0142] Además, en algunos ejemplos, un codificador de vídeo puede seleccionar el filtro de interpolación a aplicar. En dichos ejemplos, la selección del filtro puede depender de los tamaños de bloque y/o la posición relativa de las muestras dentro del bloque. Por lo tanto, en dichos ejemplos, el codificador de vídeo puede seleccionar el filtro de interpolación intra basándose en, al menos, un tamaño de bloque o posición relativa de muestra dentro del bloque. Por ejemplo, suponiendo que se haya diseñado un conjunto de filtros de interpolación diferentes, a saber $f_1, f_2, \dots, f_y$, se pueden usar diferentes filtros de interpolación para diferentes posiciones de píxel dentro del bloque actual. Por ejemplo, el filtro de interpolación f_1 puede utilizarse para las dos primeras filas de píxeles dentro del bloque actual, y el filtro de interpolación f_2 puede utilizarse para las filas de píxeles restantes.

[0143] Pueden estar disponibles diferentes filtros de interpolación para la intrapredicción de un solo bloque, y la selección del filtro de interpolación puede señalizarse explícitamente u obtenerse a partir de la información

descodificada vecina, incluidos los valores de muestra reconstruidos y los modos de intrapredicción. Por lo tanto, en dichos ejemplos, el codificador de vídeo puede seleccionar el filtro de interpolación intra basándose en al menos uno de los valores de muestra reconstruidos o modos de intrapredicción de bloques vecinos. Por ejemplo, suponiendo que se haya diseñado un conjunto de filtros de interpolación diferentes, a saber $f_1, f_2, \dots, f_y$, se pueden usar diferentes filtros de interpolación para diferentes modos de intrapredicción. Por ejemplo, el filtro de interpolación f_1 puede utilizarse para el modo de intrapredicción diagonal, y el filtro de interpolación f_2 puede utilizarse para el modo de intrapredicción vertical.

[0144] De acuerdo con algunos ejemplos de esta divulgación, se pueden permitir múltiples direcciones de intrapredicción para una PU sin señalización adicional de múltiples modos de intrapredicción. Por ejemplo, el descodificador de vídeo 30 puede descodificar información sintáctica que indica un modo de intrapredicción seleccionado para un bloque de datos de vídeo (por ejemplo, una PU) de entre una pluralidad de modos de intrapredicción. Además, el descodificador de vídeo 30 puede determinar uno o más modos de intrapredicción adicionales para el bloque sin descodificar información sintáctica adicional que indique modos de intrapredicción adicionales. En este ejemplo, el descodificador de vídeo 30 puede reconstruir el bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado y el uno o más modos de intrapredicción adicionales. Por ejemplo, el descodificador de vídeo 30 puede reconstruir una CU que contiene el bloque basándose en datos residuales para la CU y un bloque predictivo para el bloque generado basándose en el modo de intrapredicción seleccionado y el uno o más modos de intrapredicción adicionales.

[0145] En algunos ejemplos en los que se permiten múltiples direcciones de intrapredicción para una PU sin señalización adicional de múltiples modos de intrapredicción, la PU comprende una pluralidad de subbloques y cada subbloque dentro de la PU puede tener su propio modo de intrapredicción. En este ejemplo, un subbloque dentro de una PU es un bloque 4×4 . De esta manera, en un ejemplo, como parte de la determinación del uno o más modos de intrapredicción adicionales para el bloque de datos de vídeo mencionados anteriormente, el descodificador de vídeo 30 puede determinar modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo sin descodificar información sintáctica adicional que indica modos de intrapredicción adicionales. Además, en este ejemplo, como parte de la reconstrucción del bloque de datos de vídeo, el descodificador de vídeo 30 puede reconstruir los subbloques de acuerdo con los modos de intrapredicción determinados respectivos.

[0146] Además, en algunos ejemplos, la dirección de intrapredicción para un subbloque puede obtenerse como el modo de intrapredicción del bloque de referencia superior o izquierdo más cercano. De esta manera, como parte de la determinación de los modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo, el descodificador de vídeo 30 puede obtener un modo de intrapredicción respectivo para al menos uno de los subbloques como un modo de intrapredicción de un bloque de referencia superior o izquierdo más cercano. En algunos ejemplos, para cada subbloque de la PU actual, la predicción es una suma ponderada de dos direcciones de predicción procedentes del bloque más cercano en la fila de referencia superior y la columna de referencia izquierda. De esta manera, como parte de la determinación de los modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo, el descodificador de vídeo 30 puede obtener un modo de intrapredicción respectivo para al menos uno de los subbloques basándose en uno o ambos de un bloque de referencia superior y un bloque de referencia izquierdo.

[0147] De forma similar, como parte de un proceso de codificación de datos de vídeo, el codificador de vídeo 20 puede codificar información sintáctica que indica un modo de intrapredicción seleccionado para un bloque de datos de vídeo (por ejemplo, una PU) de entre una pluralidad de modos de intrapredicción. El codificador de vídeo 20 puede determinar uno o más modos de intrapredicción adicionales para el bloque de datos de vídeo sin codificar información sintáctica adicional que indique modos de intrapredicción adicionales. El codificador de vídeo 20 puede codificar el bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado y el uno o más modos de intrapredicción adicionales. En algunos ejemplos, como parte de la determinación de uno o más modos de intrapredicción adicionales para el bloque de datos de vídeo, el codificador de vídeo 20 puede determinar modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo sin codificar información sintáctica adicional que indique modos de intrapredicción adicionales y, como parte de la codificación del bloque de datos de vídeo, el codificador de vídeo 20 puede codificar los subbloques de acuerdo con los modos de intrapredicción determinados respectivos. En algunos ejemplos, como parte de la determinación de los modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo, el codificador de vídeo 20 puede obtener un modo de intrapredicción respectivo para al menos uno de los subbloques como un modo de intrapredicción de un bloque de referencia superior o izquierdo más cercano. En algunos ejemplos, como parte de la determinación de los modos de intrapredicción respectivos para cada uno de una pluralidad de subbloques del bloque de datos de vídeo, el codificador de vídeo 20 puede obtener un modo de intrapredicción respectivo para al menos uno de los subbloques basándose en uno o ambos de un bloque de referencia superior y un bloque de referencia izquierdo.

[0148] Como se mencionó anteriormente, el codificador de vídeo 20 puede realizar un proceso para decidir qué modo de intrapredicción seleccionar para una PU. Debido a que realizar un proceso completo de optimización de velocidad-distorsión para todos los 35 modos de intrapredicción disponibles en HEVC puede ser inviable en algunos casos, el codificador de vídeo 20 puede determinar una lista de N candidatos a modo de intrapredicción verificando

los modos de intrapredicción usando un criterio de "suma de diferencias de transformada absolutas" (SATD). Cuando hay disponibles más de 35 modos de intrapredicción, como es el caso en algunos ejemplos de esta divulgación, incluso el verificar los modos de intrapredicción usando el criterio SATD puede consumir mucho tiempo o recursos.

[0149] Por lo tanto, de acuerdo con un ejemplo de esta divulgación, para reducir, por ejemplo, reducir en gran medida el número de verificaciones SATD para la intrapredicción angular en el codificador de vídeo 20, se propone un régimen de verificación SATD de K fases. En este ejemplo, K es mayor que 2. En este ejemplo, para cada fase de la verificación SATD, solo un número seleccionado de modos de intrapredicción, que provienen de un subconjunto predefinido de ángulos de intrapredicción, pueden verificarse con el coste SATD. Por ejemplo, en una primera fase de verificación SATD, el codificador de vídeo 20 puede verificar cada uno de los ángulos de intrapredicción del subconjunto predefinido de modos de intrapredicción con coste SATD, y puede decidir los N mejores modos de intrapredicción. En este ejemplo, N es un valor constante predefinido que depende del tamaño de bloque. Después, en cada etapa restante de la verificación SATD, el codificador de vídeo 20 solo verifica los ángulos de intrapredicción que se aproximan, en lo que respecta a la diferencia de ángulos de predicción, a los N mejores modos de intrapredicción decididos en la fase anterior con coste SATD, y el codificador de vídeo 20 decide los N mejores modos de intrapredicción para la siguiente fase.

[0150] En un ejemplo, cuando se aplican los 65 modos angulares propuestos, el codificador de vídeo 20 aplica un esquema de verificación SATD de dos fases para decidir cuál de los modos de intrapredicción será examinado más a fondo con el criterio de coste más eficiente (por ejemplo, coste de velocidad-distorsión (R-D)) en el lado del codificador. En la primera fase, el codificador de vídeo 20 verifica solo los 35 modos de intrapredicción originales definidos en HEVC, incluidos los modos planar, DC y los 33 modos angulares, con coste SATD; a continuación se deciden los N mejores modos de intrapredicción (N puede tener el mismo valor que el usado en el software de referencia HEVC). En la segunda fase, el codificador de vídeo 20 verifica además solo los intramodos angulares que son los vecinos directos (índice de modo angular ± 1) de, es decir, contiguos a, los N mejores modos angulares con coste SATD. El codificador de vídeo 20 puede decidir verificar adicionalmente los N mejores modos de intrapredicción finales con un coste de RD.

[0151] Por lo tanto, en algunos ejemplos, el codificador de vídeo 20 puede determinar un coste SATD respectivo para codificar el bloque de datos de vídeo de acuerdo con cada uno de un primer subconjunto de modos de intrapredicción. Además, el codificador de vídeo 20 puede determinar un segundo subconjunto de los modos de intrapredicción en función de los costes SATD determinados del primer subconjunto de modos de intrapredicción. El codificador de vídeo 20 puede determinar un coste SATD respectivo para codificar el bloque de datos de vídeo de acuerdo con cada uno del segundo subconjunto de modos de intrapredicción. Además, el codificador de vídeo 20 puede seleccionar uno de los modos de intrapredicción basándose en los costes SATD determinados del primer y segundo subconjuntos de modos de intrapredicción. El codificador de vídeo 20 puede codificar el bloque de datos de vídeo de acuerdo con el seleccionado de los modos de intrapredicción.

[0152] En este ejemplo, como parte de la determinación del segundo subconjunto de los modos de intrapredicción en base a los costes SATD determinados del primer subconjunto de modos de intrapredicción, el codificador de vídeo 20 puede identificar N modos de intrapredicción del primer subconjunto de modos de intrapredicción que tienen el valor N más bajo de los costes SATD determinados. El codificador de vídeo 20 puede incluir modos de intrapredicción que tienen ángulos de intrapredicción próximos a los ángulos de intrapredicción de los N modos de intrapredicción identificados en el segundo subconjunto de modos de intrapredicción. Como parte de la inclusión de modos de intrapredicción que tienen ángulos de intrapredicción aproximados a los ángulos de intrapredicción de los N modos de intrapredicción identificados en el segundo subconjunto de modos de intrapredicción, el codificador de vídeo 20 puede incluir modos de intrapredicción que tengan ángulos de intrapredicción contiguos a ángulos de intrapredicción de los N modos de intrapredicción identificados en el segundo subconjunto de modos de intrapredicción.

[0153] Además, en algunos ejemplos, como parte de la selección de uno de los modos de intrapredicción en base a los costes SATD determinados del primer y segundo subconjuntos de modos de intrapredicción, el codificador de vídeo 20 puede identificar N modos de intrapredicción del primer y segundo subconjuntos de modos de intrapredicción que tienen el valor N más bajo de los costes SATD determinados. En este ejemplo, el codificador de vídeo 20 puede determinar un coste de velocidad-distorsión respectivo para cada uno de los N modos de intrapredicción identificados. Además, en este ejemplo, el codificador de vídeo 20 puede seleccionar uno de los N modos de intrapredicción identificados en base a los costes de velocidad-distorsión determinados.

[0154] En los ejemplos de esta divulgación relacionados con las verificaciones SATD, el valor de N depende del tamaño de bloque del bloque de datos de vídeo. Además, en algunos de dichos ejemplos, el primer subconjunto de modos de intrapredicción puede incluir un modo de intrapredicción planar, un modo de intrapredicción DC y 33 modos de predicción angulares.

[0155] En algunos ejemplos de esta divulgación relacionados con las verificaciones SATD, el primer y segundo subconjuntos de modos de intrapredicción comprenden un primer y un segundo subconjuntos de K subconjuntos de modos de intrapredicción y K es mayor que o igual a 2. En dichos ejemplos, para cada subconjunto actual de los K subconjuntos de modos de intrapredicción, el codificador de vídeo 20 puede determinar un coste SATD respectivo para codificar el bloque de datos de vídeo de acuerdo con cada modo de intrapredicción del subconjunto actual de

modos de intrapredicción. Además, el codificador de vídeo 20 puede determinar un siguiente subconjunto de los K subconjuntos de modos de intrapredicción basándose en los costes SATD determinados del subconjunto actual de modos de intrapredicción.

[0156] La FIG. 6 es un diagrama de bloques que ilustra un codificador de vídeo 20 de ejemplo que puede implementar las técnicas de esta divulgación. La FIG. 6 se proporciona con propósitos explicativos y no se debería considerar limitante de las técnicas ampliamente ejemplificadas y descritas en esta divulgación. Las técnicas de esta divulgación pueden ser aplicables a cualquiera de una variedad de normas o procedimientos de codificación.

[0157] En el ejemplo de la FIG. 6, el codificador de vídeo 20 incluye una unidad de procesamiento de predicción 100, una memoria de datos de vídeo 101, una unidad de generación residual 102, una unidad de procesamiento de transformada 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de procesamiento de transformada inversa 110, una unidad de reconstrucción 112, una unidad de filtro 114, una memoria intermedia de imágenes descodificadas 116 y una unidad de codificación por entropía 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de interpredicción 120 y una unidad de procesamiento de intrapredicción 126. La unidad de procesamiento de interpredicción 120 incluye una unidad de estimación de movimiento (ME) 122 y una unidad de compensación de movimiento (MC) 124. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0158] La memoria de datos de vídeo 101 puede almacenar datos de vídeo a codificar mediante los componentes del codificador de vídeo 20. Los datos de vídeo almacenados en la memoria de datos de vídeo 101 se pueden obtener, por ejemplo, de la fuente de vídeo 18. La memoria intermedia de imágenes descodificadas 116 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para su uso en la codificación de datos de vídeo mediante el codificador de vídeo 20, por ejemplo, en los modos de intracodificación o intercodificación. La memoria de datos de vídeo 101 y la memoria intermedia de imágenes descodificadas 116 pueden estar formadas por cualquiera de una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. El mismo dispositivo de memoria o diferentes dispositivos de memoria pueden proporcionar una memoria de datos de vídeo 101 y una memoria intermedia de imágenes descodificadas 116.

[0159] El codificador de vídeo 20 puede recibir datos de vídeo. El codificador de vídeo 20 puede codificar cada CTU en un fragmento de una imagen de los datos de vídeo. Cada una de las CTU puede estar asociada a bloques de árbol de codificación (CTB) de luma de igual tamaño y a CTB correspondientes de la imagen. Como parte de la codificación de una CTU, la unidad de procesamiento de predicción 100 puede realizar una división de árbol cuaternario para dividir los CTB de la CTU en bloques progresivamente más pequeños. Los bloques más pequeños pueden ser bloques de codificación de CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un CTB asociado a una CTU en cuatro subbloques de igual tamaño, dividir uno o más de los subbloques en cuatro subsubbloques de igual tamaño, y así sucesivamente.

[0160] El codificador de vídeo 20 puede codificar las CU de una CTU para generar representaciones codificadas de las CU (es decir, CU codificadas). Como parte de la codificación de una CU, la unidad de procesamiento de predicción 100 puede dividir los bloques de codificación asociados a la CU entre una o más PU de la CU. Por tanto, cada PU puede estar asociada a un bloque de predicción de luma y a bloques de predicción de croma correspondientes. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden admitir PU que tienen diversos tamaños. Como se indica anteriormente, el tamaño de una CU se puede referir al tamaño del bloque de codificación de luma de la CU, y el tamaño de una PU se puede referir al tamaño de un bloque de predicción de luma de la PU. Suponiendo que el tamaño de una CU particular es de $2N \times 2N$, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden admitir tamaños de PU de $2N \times 2N$ o $N \times N$ para la intrapredicción, y tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$, o similares, para la interpredicción. El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden admitir una división asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$ para la interpredicción.

[0161] La unidad de procesamiento de interpredicción 120 puede generar datos predictivos para una PU realizando una interpredicción en cada PU de una CU. Los datos predictivos para la PU pueden incluir bloques predictivos de la PU e información de movimiento para la PU. La unidad de procesamiento de intrapredicción 126 puede generar datos predictivos para una PU realizando intrapredicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos para la PU y diversos elementos sintácticos. La unidad de procesamiento de intrapredicción 126 puede realizar intrapredicción en las PU en fragmentos I, fragmentos P y fragmentos B.

[0162] Para realizar la intrapredicción en una PU, la unidad de procesamiento de intrapredicción 126 puede usar múltiples modos de intrapredicción para generar múltiples conjuntos de bloques predictivos para la PU. Cuando se realiza la intrapredicción usando un modo particular de intrapredicción, la unidad de procesamiento de intrapredicción 126 puede generar bloques predictivos para la PU usando un conjunto particular de muestras de bloques contiguos. Los bloques contiguos pueden estar arriba, arriba y a la derecha, arriba y a la izquierda, o a la izquierda de los bloques de predicción de la PU, suponiendo un orden de codificación de izquierda a derecha y de arriba a abajo para las PU, CU y CTU. La unidad de procesamiento de intrapredicción 126 puede usar diversos números de modos de intrapredicción, por ejemplo, modos de predicción DC, planares o direccionales/angulares, como se describe en el

presente documento. En algunos ejemplos, el número de modos de intrapredicción puede depender del tamaño de los bloques de predicción de la PU.

[0163] La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para las PU de una CU de entre los datos predictivos generados por la unidad de procesamiento de interpretación 120 para las PU, o los datos predictivos generados por la unidad de procesamiento de intrapredicción 126 para las PU. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos predictivos para las PU de la CU en base a unas métricas de velocidad/distorsión de los conjuntos de datos predictivos. Los bloques predictivos de los datos predictivos seleccionados pueden denominarse en el presente documento bloques predictivos seleccionados.

[0164] La unidad de generación residual 102 puede generar, basándose en el bloque de codificación de luma, Cb y Cr de una CU y los bloques predictivos seleccionados de luma, Cb y Cr de las PU de la CU, bloques residuales de luma, Cb y Cr de la CU. Por ejemplo, la unidad de generación residual 102 puede generar los bloques residuales de la CU de manera que cada muestra en los bloques residuales tenga un valor igual a una diferencia entre una muestra en un bloque de codificación de la CU y una muestra correspondiente en un bloque predictivo seleccionado correspondiente de una PU de la CU.

[0165] La unidad de procesamiento de transformada 104 puede realizar división de árbol cuaternario para dividir los bloques residuales de una CU en bloques de transformada asociados a las TU de la CU. De este modo, una TU puede estar asociada a un bloque de transformada de luma y dos bloques de transformada de croma correspondientes. Los tamaños y las posiciones de los bloques de transformada de luma y croma de las TU de una CU pueden o no estar basados en los tamaños y las posiciones de bloques de predicción de las PU de la CU.

[0166] La unidad de procesamiento de transformada 104 puede generar bloques de coeficientes de transformada para cada TU de una CU aplicando una o más transformadas a los bloques de transformada de la TU. La unidad de procesamiento de transformada 104 puede aplicar diversas transformadas a un bloque de transformada asociado a una TU. Por ejemplo, la unidad de procesamiento de transformada 104 puede aplicar una transformada de coseno discreta (DCT), una transformada direccional o una transformada conceptualmente similar a un bloque de transformada. En algunos ejemplos, la unidad de procesamiento de transformada 104 no aplica transformadas a un bloque de transformada. En dichos ejemplos, el bloque de transformada se puede tratar como un bloque de coeficientes de transformada.

[0167] La unidad de cuantificación 106 puede cuantificar los coeficientes de transformada en un bloque de coeficientes. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes de transformada. Por ejemplo, un coeficiente de transformada de n bits puede redondearse por defecto hasta un coeficiente de transformada de m bits durante la cuantificación, donde n es mayor que m . La unidad de cuantificación 106 puede cuantificar un bloque de coeficientes asociado a una TU de una CU basándose en un valor de parámetro de cuantificación (QP) asociado a la CU. El codificador de vídeo 20 puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes asociados a una CU, ajustando el valor QP asociado a la CU. La cuantificación puede introducir pérdida de información, por lo que los coeficientes de transformada cuantificados pueden tener una precisión menor que los originales.

[0168] La unidad de cuantificación inversa 108 y la unidad de procesamiento de transformada inversa 110 pueden aplicar una cuantificación inversa y transformadas inversas a un bloque de coeficientes, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficientes. La unidad de reconstrucción 112 puede añadir el bloque residual reconstruido a las muestras correspondientes de uno o más bloques predictivos generados por la unidad de procesamiento de predicción 100 para generar un bloque de transformada reconstruido asociado a una TU. Reconstruyendo bloques de transformada para cada TU de una CU de esta manera, el codificador de vídeo 20 puede reconstruir los bloques de codificación de la CU.

[0169] La unidad de filtro 114 puede realizar una o más operaciones de reducción de efecto bloque para reducir los artefactos de efecto bloque en los bloques de codificación asociados a una CU. La memoria intermedia de imágenes descodificadas 116 puede almacenar los bloques de codificación reconstruidos después de que la unidad de filtro 114 realice las una o más operaciones de reducción de efecto bloque en los bloques de codificación reconstruidos. La unidad de procesamiento de interpretación 120 puede usar una imagen de referencia que contiene los bloques de codificación reconstruidos para realizar una interpretación en las PU de otras imágenes. Además, la unidad de procesamiento de intrapredicción 126 puede usar bloques de codificación reconstruidos de la memoria intermedia de imágenes descodificadas 116 para realizar una intrapredicción en otras PU de la misma imagen que la CU.

[0170] La unidad de codificación por entropía 118 puede recibir datos desde otros componentes funcionales del codificador de vídeo 20. Por ejemplo, la unidad de codificación por entropía 118 puede recibir bloques de coeficientes desde la unidad de cuantificación 106 y puede recibir elementos sintácticos desde la unidad de procesamiento de predicción 100. La unidad de codificación por entropía 118 puede realizar una o más operaciones de codificación por entropía en los datos para generar datos codificados por entropía. Por ejemplo, la unidad de codificación por entropía 118 puede realizar una operación de codificación de longitud variable adaptativa al contexto (CAVLC), una operación CABAC, una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética

binaria adaptativa al contexto basada en sintaxis (SBAC), una operación de codificación por entropía de división en intervalos de probabilidad (PIPE), una operación de codificación Golomb exponencial u otro tipo de operación de codificación por entropía en los datos. El codificador de vídeo 20 puede proporcionar un flujo de bits que incluye datos codificados por entropía generados por la unidad de codificación por entropía 118.

[0171] El codificador de vídeo 20 es un ejemplo de un codificador de vídeo configurado para realizar cualquiera de las técnicas descritas en esta divulgación. De acuerdo con una o más técnicas de esta divulgación, una o más unidades del codificador de vídeo 20 pueden realizar las técnicas descritas en el presente documento como parte de un proceso de codificación de vídeo. Por ejemplo, la unidad de procesamiento de intrapredicción 126 y/o la unidad de codificación por entropía 118 pueden configurarse para realizar las técnicas descritas en el presente documento como parte de un proceso de codificación de vídeo. En algunos ejemplos, la unidad de procesamiento de intrapredicción 126 puede configurarse para realizar las técnicas descritas en el presente documento para obtener o seleccionar modos de intrapredicción y/o MPM para intrapredicción, las técnicas que usan un mayor número de MPM y/o modos angulares y las técnicas para aplicar un filtro de interpolación intra de N etapas, donde N es mayor que 2. La unidad de procesamiento de intrapredicción 126 puede configurarse para realizar las técnicas descritas en el presente documento para permitir múltiples direcciones de intrapredicción por bloque, por ejemplo, direcciones respectivas para subbloques del bloque, que pueden no requerir señalización adicional de múltiples modos de intrapredicción.

[0172] La unidad de procesamiento de intrapredicción 126 puede proporcionar información sintáctica a la unidad de codificación por entropía 118, tal como un modo de intrapredicción seleccionado y/o un índice MPM para intrapredicción. La unidad de procesamiento de intrapredicción 126 y/o la unidad de codificación por entropía 118 pueden configurarse para implementar las técnicas descritas en el presente documento para codificar o señalar la información sintáctica relacionada con la intrapredicción, por ejemplo, el modo de intrapredicción seleccionado y/o el índice MPM para intrapredicción, en un flujo de bits de vídeo codificado, que puede ser descodificado por el descodificador de vídeo 30.

[0173] La FIG. 7 es un diagrama de bloques que ilustra un descodificador de vídeo 30 de ejemplo que está configurado para implementar las técnicas de esta divulgación. La FIG. 7 se proporciona con propósitos explicativos y no se limita a las técnicas ampliamente ejemplificadas y descritas en esta divulgación. Las técnicas de esta divulgación pueden ser aplicables a una variedad de normas o procedimientos de codificación.

[0174] En el ejemplo de la FIG. 7, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 150, una memoria de datos de vídeo 151, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de procesamiento de transformada inversa 156, una unidad de reconstrucción 158, una unidad de filtro 160 y una memoria intermedia de imágenes descodificadas 162. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 164 y una unidad de procesamiento de intrapredicción 166. En otros ejemplos, el descodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

[0175] La memoria de datos de vídeo 151 puede almacenar datos de vídeo, tales como un flujo de bits de vídeo codificado que los componentes del descodificador de vídeo 30 van a descodificar. Los datos de vídeo almacenados en la memoria de datos de vídeo 151 pueden obtenerse, por ejemplo, a partir de un medio legible por ordenador 16, por ejemplo, desde una fuente de vídeo local, tal como una cámara, mediante comunicación de datos de vídeo por red alámbrica o inalámbrica, o accediendo a medios de almacenamiento físico de datos. La memoria de datos de vídeo 151 puede formar una memoria intermedia de imágenes codificadas (CPB) que almacena datos de vídeo codificados a partir de un flujo de bits de vídeo codificado. La memoria intermedia de imágenes descodificadas 162 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para su uso en la descodificación de datos de vídeo mediante el descodificador de vídeo 30, por ejemplo, en los modos de intracodificación o intercodificación. La memoria de datos de vídeo 151 y la memoria intermedia de imágenes descodificadas 162 pueden estar formadas por cualquiera de una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. El mismo dispositivo de memoria o diferentes dispositivos de memoria pueden proporcionar una memoria de datos de vídeo 151 y una memoria intermedia de imágenes descodificadas 162.

[0176] Una memoria intermedia de imágenes codificadas (CPB) formada por una memoria de datos de vídeo 151 puede recibir y almacenar datos de vídeo codificados (por ejemplo, unidades NAL) de un flujo de bits. La unidad de descodificación por entropía 150 puede recibir unidades NAL de la CPB formada por la memoria de datos de vídeo 151 y analizar sintácticamente las unidades NAL para obtener elementos sintácticos a partir del flujo de bits. La unidad de descodificación por entropía 150 puede descodificar por entropía elementos sintácticos codificados por entropía en las unidades NAL. La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de procesamiento de transformada inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 160 pueden generar datos de vídeo descodificados en base a los elementos sintácticos extraídos del flujo de bits.

[0177] Las unidades NAL del flujo de bits pueden incluir unidades NAL de fragmentos codificados. Como parte de la descodificación del flujo de bits, la unidad de descodificación por entropía 150 puede extraer y descodificar por

entropía elementos sintácticos de las unidades NAL de fragmentos codificados. Cada uno de los fragmentos codificados puede incluir una cabecera de fragmento y datos de fragmento. La cabecera de fragmento puede contener elementos sintácticos pertenecientes a un fragmento.

5 **[0178]** Además de descodificar elementos sintácticos del flujo de bits, el descodificador de vídeo 30 puede realizar una operación de descodificación en una CU. Realizando la operación de descodificación en una CU, el descodificador de vídeo 30 puede reconstruir bloques de codificación de la CU.

10 **[0179]** Como parte de la realización de una operación de descodificación en una CU, la unidad de cuantificación inversa 154 puede cuantificar de forma inversa, es decir, descuantificar, los bloques de coeficientes asociados a las TU de la CU. La unidad de cuantificación inversa 154 puede usar un valor QP asociado a la CU de la TU para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa para que la unidad de cuantificación inversa 154 lo aplique. Es decir, la relación de compresión, es decir, la relación entre el número de bits usados para representar la secuencia original y la comprimida, puede controlarse ajustando el valor del QP usado al cuantificar los coeficientes de transformada. La relación de compresión también puede depender del procedimiento de codificación por entropía empleado.

20 **[0180]** Después de que la unidad de cuantificación inversa 154 haya realizado la cuantificación inversa de un bloque de coeficientes, la unidad de procesamiento de transformada inversa 156 puede aplicar una o más transformadas inversas al bloque de coeficientes para generar un bloque residual asociado a la TU. Por ejemplo, la unidad de procesamiento de transformada inversa 156 puede aplicar al bloque de coeficientes una DCT inversa, una transformada entera inversa, una transformada inversa de Karhunen-Loeve (KLT), una transformada de rotación inversa, una transformada direccional inversa u otra transformada inversa.

25 **[0181]** Si se codifica una PU usando intrapredicción, la unidad de procesamiento de intrapredicción 166 puede realizar una intrapredicción para generar bloques predictivos para la PU. La unidad de procesamiento de intrapredicción 166 puede usar un modo de intrapredicción para generar los bloques predictivos de luma, Cb y Cr para la PU basándose en los valores de muestra reconstruidos de bloques vecinos espacialmente. La unidad de procesamiento de intrapredicción 166 puede determinar el modo de intrapredicción para la PU basándose en uno o más elementos sintácticos descodificados a partir del flujo de bits.

30 **[0182]** La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (RefPicList0) y una segunda lista de imágenes de referencia (RefPicList1) basándose en elementos sintácticos extraídos del flujo de bits. Además, si se codifica una PU usando interpredicción, la unidad de descodificación por entropía 150 puede obtener información de movimiento para la PU. La unidad de compensación de movimiento 164 puede determinar, basándose en la información de movimiento de la PU, una o más regiones de referencia para la PU. La unidad de compensación de movimiento 164 puede generar, basándose en muestras en los uno o más bloques de referencia para la PU, bloques predictivos de luma, Cb y Cr para la PU.

40 **[0183]** La unidad de reconstrucción 158 puede usar los valores residuales de los bloques de transformada de luma, Cb y Cr asociados a las TU de una CU y los bloques predictivos de luma, Cb y Cr de las PU de la CU, es decir, datos de intrapredicción o datos de interpredicción, según corresponda, para reconstruir los bloques de codificación de luma, Cb y Cr de la CU. Por ejemplo, la unidad de reconstrucción 158 puede añadir muestras de los bloques de transformada de luma, Cb y Cr a muestras correspondientes de los bloques predictivos de luma, Cb y Cr para reconstruir los bloques de codificación de luma, Cb y Cr de la CU.

45 **[0184]** La unidad de filtro 160 puede realizar una operación de reducción del efecto bloque para reducir los artefactos de efecto bloque asociados a los bloques de codificación de luma, Cb y Cr de la CU. El descodificador de vídeo 30 puede almacenar los bloques de codificación de luma, Cb y Cr de la CU en la memoria intermedia de imágenes descodificadas 162. La memoria intermedia de imágenes descodificadas 162 puede proporcionar imágenes de referencia para una posterior compensación de movimiento, intrapredicción y presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Por ejemplo, el descodificador de vídeo 30 puede realizar, basándose en los bloques de luma, Cb y Cr de la memoria intermedia de imágenes descodificadas 162, operaciones de intrapredicción o de interpredicción en las PU de otras CU. De esta manera, el descodificador de vídeo 50 30 puede extraer, del flujo de bits, niveles de coeficientes de transformada del bloque de coeficientes de luma significativos, realizar la cuantificación inversa de los niveles de coeficientes de transformada, aplicar una transformada a los niveles de coeficientes de transformada para generar un bloque de transformada, generar en base a, al menos en parte, el bloque de transformada, un bloque de codificación, y proporcionar el bloque de codificación para su visualización.

60 **[0185]** El descodificador de vídeo 30 es un ejemplo de un descodificador de vídeo configurado para realizar cualquiera de las técnicas descritas en esta divulgación. De acuerdo con una o más técnicas de esta divulgación, una o más unidades del descodificador de vídeo 30 pueden realizar las técnicas descritas en el presente documento como parte de un proceso de descodificación de vídeo. Por ejemplo, la unidad de procesamiento de intrapredicción 166 y/o la unidad de descodificación por entropía 150 pueden configurarse para realizar las técnicas descritas en el presente documento como parte de un proceso de descodificación de vídeo. En algunos ejemplos, la unidad de procesamiento 65

de intrapredicción 166 puede configurarse para realizar las técnicas descritas en el presente documento para obtener o seleccionar modos de intrapredicción y/o MPM para intrapredicción, las técnicas que usan un mayor número de MPM y/o modos angulares y las técnicas para aplicar un filtro de interpolación intra de N etapas, donde N es mayor que 2. La unidad de procesamiento de intrapredicción 166 puede configurarse para realizar las técnicas descritas en el presente documento para permitir múltiples direcciones de intrapredicción por bloque, por ejemplo, direcciones respectivas para subbloques del bloque, que pueden no requerir señalización adicional de múltiples modos de intrapredicción.

[0186] La unidad de procesamiento de intrapredicción 166 puede recibir información sintáctica desde la unidad de descodificación por entropía 150, tal como un modo de intrapredicción seleccionado y/o un índice MPM para intrapredicción. La unidad de procesamiento de intrapredicción 166 y/o la unidad de descodificación por entropía 150 pueden configurarse para implementar las técnicas descritas en el presente documento para descodificar la información sintáctica relacionada con la intrapredicción, por ejemplo, el modo de intrapredicción seleccionado y/o el índice MPM para intrapredicción, de un flujo de bits de vídeo codificado.

[0187] En las subsecciones siguientes se proporcionan implementaciones de ejemplo de técnicas particulares de esta divulgación. En la práctica, cualquier combinación de cualquier parte de los ejemplos puede usarse como un nuevo ejemplo. Las implementaciones de ejemplo se muestran en forma de ediciones a la especificación JCTVC-N1003. A continuación, el texto añadido a la especificación JCTVC-N1003 se muestra entre etiquetas <ins> y </ins>, donde "ins" es la abreviatura de "insert" (insertar). El texto eliminado de la especificación JCTVC-N1003 se muestra entre etiquetas y , donde "dlt" es la abreviatura de "delete" (borrar).

[0188] Anteriormente se describieron varios ejemplos en los que el codificador de vídeo 20 y el descodificador de vídeo 30 usan más de tres MPM para señalar un modo de intrapredicción de una PU. En algunos de dichos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 usan codificación CABAC para codificar y descodificar índices MPM. Como parte de la codificación CABAC, el codificador de vídeo 20 aplica un proceso de binarización a un índice MPM para convertir el índice MPM en un código binario y el descodificador de vídeo 30 desbinariza el código binario para recuperar el índice MPM. De acuerdo con una técnica de esta divulgación, como se muestra a continuación en la Tabla 9-32, el índice MPM (mpm_idx) se binariza usando un proceso de binarización de Rice truncado (TR) y el parámetro de entrada cMax se cambia de 2 a 5 para permitir valores más grandes del índice MPM. La sección 9.3.3.2 de JCTVC-N1003 describe una implementación del proceso de binarización de Rice truncado.

Tabla 9-32 - Elementos sintácticos y binarizaciones asociadas

Elemento sintáctico	Binarización	
	Proceso	Parámetros de entrada
...	...	...
cu_transquant_by_pass_flag	FL	cMax = 1
cu_skip_flag	FL	cMax = 1
pred_mode_flag	FL	cMax = 1
part_mode	9.3.3.5	(xCb, yCb) = (x0, y0), log2CbSize
pcm_flag[][]	FL	cMax = 1
prev_intra_luma_pred_flag[][]	FL	cMax = 1
mpm_idx[][]	TR	cMax = 2<ins>5</ins>, cRiceParam = 0
rem_intra_luma_pred_mode[][]	FL	cMax = 31
intra_chroma_pred_mode[][]	9.3.3.6	-
rqt_root_cbf	FL	cMax = 1

[0189] Además, como se describe anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar modelado de contexto cuando codifican (por ejemplo, codifican mediante CABAC) uno o más *bins* de un índice MPM. En JCTVC-N1003, un índice MPM binarizado (es decir, mpm_idx) consiste en tres *bins*. Sin embargo, en esta divulgación, con un mayor número de MPM, el índice MPM binarizado puede incluir cinco o más *bins*. Como se especifica en la subcláusula 9.3.4.2.1 de JCTVC-N1003, una variable ctxInc se especifica mediante la entrada correspondiente en la Tabla 9-37, y cuando se enumera más de un valor en la Tabla 9-37 para binIdx, el proceso de asignación para ctxInc para ese binIdx se especifica adicionalmente en las subcláusulas dadas entre paréntesis. Como se describe más detalladamente en la subcláusula 9.3.4.2.1. de JCTVC-N1003, ctxInc se usa para determinar un valor de ctxIdx, que es un índice de un contexto de codificación. Como se muestra a continuación en la versión editada de la Tabla 9-37, los tres primeros *bins* del índice MPM pueden codificarse usando contextos de codificación seleccionados usando los procedimientos de modelado de contexto de la siguiente subcláusula 9.3.4.x.x añadida, y los *bins* restantes del índice MPM pueden codificarse usando codificación de derivación.

Tabla 9-37 - Asignación de ctxInc a elementos sintácticos con bins codificados por contexto

Elemento sintáctico	binIdx					
	0	1	2	3	4	>= 5
...	...	...	...	...	...	...
pcm_flag[][]	terminar	no aplicable	no aplicable	no aplicable	no aplicable	no aplicable
prev_intra_luma_pred_flag[][]	0	no aplicable	no aplicable	no aplicable	no aplicable	no aplicable
mpm_idx[][]	<dlt> derivación </dlt><ins> 0,1,2 (subcláusula 9.3.4.x.x) </ins>	 derivación <ins> 3,4,5 (subcláusula 9.3.4.x.x) </ins>	 derivación <ins> 7,8 (subcláusula 9.3.4.x.x) </ins>	<dlt> no aplicable </dlt><ins> derivación </ins>	<dlt> no aplicable </dlt><ins> derivación </ins>	<dlt> no aplicable </dlt><ins> derivación </ins>
rem_intra_luma_pred_mode[][]	derivación	derivación	derivación	derivación	derivación	derivación
intra_chroma_pred_mode[][]	0	derivación	derivación	no aplicable	no aplicable	no aplicable
...	...	...	...	...	...	...

<ins>9.3.3.x Proceso de binarización para rem_intra_luma_pred_mode

[0190] La entrada a este proceso es una solicitud de una binarización para el elemento sintáctico rem_intra_luma_pred_mode.

[0191] La salida de este proceso es la binarización del elemento sintáctico.

- Si rem_intra_luma_pred_mode es igual a 28, la cadena de binarización es "111";
- De lo contrario, la binarización es un código de longitud fija con 5 bins.

9.3.4.x.x Proceso de obtención de ctxInc para el elemento sintáctico mpm_idx

[0192] Las entradas a este proceso son los intramodos del bloque vecino izquierdo candIntraPredModeA y del bloque vecino superior candIntraPredModeB.

[0193] La salida de este proceso es la variable ctxInc.

[0194] La variable csbfCtx se obtiene usando la ubicación actual (xS, yS), dos bins previamente descodificados del elemento sintáctico coded_sub_block_flag en orden de exploración, y el tamaño de bloque de transformada log2TrafoSize, como sigue:

- Cuando candIntraPredModeA es igual a candIntraPredModeB, ctxInc se obtiene como sigue ctxInc = candIntraPredModeA > 1 ? 0 : 1
- De lo contrario, ctxInc = (candIntraPredModeA && candIntraPredModeB) ? 2 : 3 </ins>

[0195] De acuerdo con algunos ejemplos de esta divulgación, se definen más de tres MPM. El siguiente texto describe ejemplos de cambios en JCTVC-N1003 para implementar seis MPM.

8.4.2 Proceso de obtención para el modo de intrapredicción de luma

[0196] La entrada para este proceso es una ubicación de luma (xPb, yPb) que especifica la muestra superior izquierda del bloque de predicción de luma actual en relación con la muestra de luma superior izquierda de la imagen actual<ins>, un tamaño de bloque de predicción nPbS</ins>.

[0197] En este proceso, se obtiene el modo de intrapredicción de luma IntraPredModeY[xPb][yPb]. La Tabla 8-1 especifica el valor para el modo de intrapredicción y los nombres asociados.

Tabla 8-1 - Especificación del modo de intrapredicción y nombres asociados

Modo de intrapredicción	Nombre asociado
0	INTRA_PLANAR
1	INTRA_DC
2..34	INTRA_ANGULAR2..INTRA_ANGULAR34

[0198] IntraPredModeY[xPb][yPb] etiquetado como 0..34 representa direcciones de predicciones como se ilustra en la Figura 8-1, que es la FIG. 8 de esta divulgación.

[0199] IntraPredModeY[xPb][yPb] se obtiene mediante las siguientes etapas ordenadas:

4. Las ubicaciones vecinas (xNbA, yNbA) y (xNbB, yNbB) se establecen iguales a (xPb - 1, yPb) y (xPb, yPb - 1), respectivamente.

5. Para que X sea reemplazada por A o B, las variables candIntraPredModeX se obtienen de la siguiente manera:

<ins>

- candIntraPredModeX se inicializa como INTRA DC.

- Inicializar los recuentos de uso de intramodo como 0: cntIntraPredModeX[i]=0, i=0, 1, ...,34

- Inicializar el recuento máximo de uso de intramodo como 0: cntMaxIntraPredModeX = 0.

- Para x = 0..nPbS - 1, se aplica lo siguiente

- (xCurr, yCurr) = (X==A) ? (xNbA+x, yNbA) ? (xNbA, yNbA +x)

- El proceso de obtención de disponibilidad para un bloque en orden de exploración en z como se especifica en la subcláusula 6.4.1 se invoca con la ubicación (xCurr,yCurr) establecida igual a (xPb, yPb) y la ubicación vecina (xNbY, yNbY) establecida igual a (xNbX, yNbX) como entradas, y la salida se asigna a availableX.

- Si availableX es igual a TRUE y CuPredMode[xNbX][yNbX] es igual a MODE INTRA, se aplica lo siguiente.

- cntIntraPredModeX[CuPredMode[xCurr][yCurr]]++

- si cntIntraPredModeX[CuPredMode[xCurr][yCurr]] > cntMaxIntraPredModeX, se aplica lo siguiente:

- cntMaxIntraPredModeX = cntIntraPredModeX[CuPredMode[xCurr][yCurr]];

- candIntraPredModeX=CuPredMode[xCurr][yCurr] </ins> <dlt>

- El proceso de obtención de disponibilidad para un bloque en orden de exploración en z como se especifica en la subcláusula 6.4.1 se invoca con la ubicación (xCurr, yCurr) establecida igual a (xPb, yPb) y la ubicación vecina (xNbY, yNbY) establecida igual a (xNbX, yNbX) como entradas, y la salida se asigna a availableX.

- El modo de intrapredicción candidato candIntraPredModeX se obtiene de la siguiente manera:

- Si availableX es igual a FALSE, candIntraPredModeX se establece igual a INTRA DC.

- De lo contrario, si CuPredMode[xNbX][yNbX] no es igual a MODE_INTRA o pcm flag[xNbX][yNbX] es igual a 1, candIntraPredModeX se establece igual a INTRA DC,

- De lo contrario, si X es igual a B e yPb - 1 es menor que ((yPb >> CtbLog2SizeY) << CtbLog2SizeY), candIntraPredModeB se establece igual a INTRA_DC.

- De lo contrario, candIntraPredModeX se establece igual a IntraPredModeY[xNbX][yNbX].</dlt>

6. candModeList[x] con x = 0..

- <ins> idxPlanar se establece en 0, idxDC se establece en 1;</ins>

- Si candIntraPredModeB es igual a candIntraPredModeA, se aplica lo siguiente:

- Si candIntraPredModeA es menor que 2 (es decir, igual a INTRA_PLANAR o INTRA_DC), candModeList[x] con x = 0..

candModeList[0] = INTRA PLANAR (8-15)

candModeList[1] = INTRA_DC (8-16)

candModeList[2] = INTRA ANGULAR26 (8-17)

~~ins~~candModeList[3] = INTRA_ANGULAR10 (8-17) candModeList[4] = INTRA_ANGULAR2 (8-17)
candModeList[5] = INTRA_ANGULAR18 (8-17)~~ins~~

- 5 - De lo contrario, candModeList[x] con $x = 0..5$ se obtiene como sigue:

candModeList[0] = candIntraPredModeA (8-18)

10 ~~dlt~~candModeList[2] = 2 + ((candIntraPredModeA + 29) % 32) (8-19) candModeList[2] = 2 + ((candIntraPredModeA - 2 + 1) % 32) (8-20)~~dlt~~

~~ins~~candModeList[1] = INTRA_PLANAR (8-20) candModeList[2] = 2 + ((candIntraPredModeA - 1) % 32) (8-19)
candModeList[3] = 2 + ((candIntraPredModeA + 29) % 32) (8-20) candModeList[4] = 2 + ((candModeList[2] - 1) % 32) (8-20) candModeList[5] = INTRA_DC (8-20) idxPlanar = 1 idxDC = 5~~ins~~

- 15 - De lo contrario (candIntraPredModeB no es igual a candIntraPredModeA), se aplica lo siguiente:

- candModeList[0] y candModeList[1] se obtienen como sigue:

20 candModeList[0] = candIntraPredModeA (8-21)

candModeList[1] = candIntraPredModeB (8-22)

- 25 - Si ninguno de candModeList[0] y candModeList[1] es igual a INTRA_PLANAR, candModeList[2] se establece igual a INTRA_PLANAR, ~~ins~~ y se aplica lo siguiente:

- maxDir = max(candModeList[0], candModeList[1])

- 30 - minDir = min(candModeList[0], candModeList[1])

- idxPlanar = 2

- Si cualquiera de candModeList[0] y candModeList[1] es igual a INTRA_DC, se aplica lo siguiente:

35 candModeList[3] = 2 + ((maxDir + 29) % 32) (8-21)

candModeList[4] = 2 + ((maxDir - 1) % 32) (8-22)

40 candModeList[5] = 2 + ((candModeList[4] - 1) % 32) (8-22) idxDC = (candModeList[0] == INTRA_DC) ? 0 : 1

- De lo contrario,

candModeList[3] = INTRA_DC (8-21)

45 candModeList[4] = 2 + ((maxDir - 1) % 32) (8-22)

- Si candModeList[4] es igual a minDir, candModeList[4]++ candModeList[5] = 2 + ((candModeList[4] + 29) % 32) (8-22)

- 50 - Si candModeList[5] es igual a maxDir, candModeList[5]--

- Si candModeList[5] es igual a candModeList[4], candModeList[5] = minDir+1 idxDC = 3~~ins~~

- 55 - De lo contrario, si ninguno de candModeList[0] y candModeList[1] es igual a INTRA_DC, candModeList[2] se establece igual a INTRA_DC, ~~ins~~ se aplica lo siguiente:

candModeList[3] = 2 + ((candIntraPredModeA + 29) % 32)

candModeList[4] = 2 + ((candIntraPredModeA - 1) % 32)

60 candModeList[5] = 2 + ((candModeList[4] - 1) % 32)

idxPlanar = (candModeList[0] == INTRA_PLANAR) ? 0 : 1 idxDC = 2~~ins~~

- 65 - De lo contrario, candModeList[2] se establece igual a INTRA_ANGULAR26, se aplica lo siguiente:

~~ins~~candModeList[3] = INTRA_ANGULAR10 candModeList[4] = INTRA_ANGULAR2 candModeList[5] = INTRA_ANGULAR18 idxPlanar = (candModeList[0]== INTRA_PLANAR) ? 0 : 1 idxDC = 1- idxPlanar~~ins~~

7. IntraPredModeY[xPb][yPb] se obtiene aplicando el siguiente procedimiento:

- Si prev_intra_luma_pred_flag[xPb][yPb] es igual a 1, IntraPredModeY[xPb][yPb] se establece igual a candModeList[mpm_idx].

- De lo contrario, IntraPredModeY[xPb][yPb] se obtiene aplicando las siguientes etapas ordenadas:

1. La matriz candModeList[x], x = 0...~~dlt>2</dlt>~~~~ins>5</ins>~~ se modifica como las siguientes etapas ordenadas:~~ins>~~

i. candModeList[idxPlanar] = candModeList[0]

ii. candModeList[idxDC] = candModeList[1]

iii. candModeList[0] = INTRA_PLANAR

iv. candModeList[1] = INTRA_DC~~ins>~~

v. Cuando candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~] es mayor que candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~], ambos valores se intercambian de la siguiente manera:

(candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~], candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~]) = Swap(candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~], candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~]) (8-23)

vi. Cuando candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~] es mayor que candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~], ambos valores se intercambian de la siguiente manera:

(candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~], candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~]) = Swap(candModeList[~~dlt>0</dlt>~~~~ins>2</ins>~~], candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~]) (8-24)

vii. ~~ins>~~ Cuando candModeList[2] es mayor que candModeList[5], ambos valores se intercambian de la siguiente manera: (candModeList[2], candModeList[5]) = Swap(candModeList[2], candModeList[5])~~ins>~~

viii. Cuando candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~] es mayor que candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~], ambos valores se intercambian de la siguiente manera:

(candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~], candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~]) = Swap(candModeList[~~dlt>1</dlt>~~~~ins>3</ins>~~], candModeList[~~dlt>2</dlt>~~~~ins>4</ins>~~]) (8-25)

ix. ~~ins>~~ Cuando candModeList[3] es mayor que candModeList[5], ambos valores se intercambian de la siguiente manera:

(candModeList[3], candModeList[5]) = Swap(candModeList[3], candModeList[5])

x. Cuando candModeList[4] es mayor que candModeList[5], ambos valores se intercambian de la siguiente manera:

(candModeList[4], candModeList[5]) = Swap(candModeList[4], candModeList[5])~~ins>~~

2. IntraPredModeY[xPb][yPb] se obtiene mediante las siguientes etapas ordenadas:

i. IntraPredModeY[xPb][yPb] se establece igual a rem_intra_luma_pred_mode[xPb][yPb].

ii. Para i igual a 0 hasta ~~dlt>2</dlt>~~~~ins>5</ins>~~, ambos inclusive, cuando IntraPredModeY[xPb][yPb] es mayor que o igual a candModeList[i], el valor de IntraPredModeY[xPb][yPb] se incrementa en uno.

[0200] En la versión adaptada de ejemplo de la subcláusula 8.4.2 de JCTVC-N1003 mostrada anteriormente, el proceso de obtención para los MPM de la PU actual incluye un proceso de obtención para un modo de intrapredicción representativo (candIntraPredModeA) para una columna vecina izquierda y un modo de intrapredicción representativo (candIntraPredModeB) para una fila vecina superior. En este ejemplo, la columna vecina izquierda corresponde a las muestras (xNbA, yNbA + x) para x = x..nPbs y la fila vecina superior corresponde a las muestras (xNbA+x, yNbA) para x = 0..nPbs, donde nPbs es el ancho y la altura del bloque de predicción de la PU actual.

[0201] Además, en la versión adaptada de ejemplo de la subcláusula 8.4.2 de JCTVC-N1003 mostrada anteriormente, la lista de candidatos MPM (es decir, candModeList) siempre incluye los modos de intrapredicción planar, DC, vertical (es decir, INTRA_ANGULAR26), horizontal (es decir, INTRA_ANGULAR10), y diagonal (es decir, INTRA_ANGULAR18) si candIntraPredModeA es igual que candIntraPredModeB y candIntraPredModeA es planar o DC. Sin embargo, en este ejemplo, si candIntraPredModeA no es planar ni DC, la lista de candidatos a MPM incluye candIntraPredModeA, la intrapredicción planar, tres modos de intrapredicción angulares más aproximados a candIntraPredModeA y el modo de intrapredicción DC. Los tres modos de intrapredicción angulares más aproximados a candIntraPredModeA se calculan como:

$$2 + ((\text{candIntraPredModeA} - 1) \% 32),$$

$$(2 + ((\text{candIntraPredModeA} + 29) \% 32), \text{ y}$$

$$2 + ((\text{candModeList}[2] - 1) \% 32)$$

En las fórmulas anteriores, % denota la operación de módulo.

[0202] De forma similar, en la versión adaptada de ejemplo de la subcláusula 8.4.2 de JCTVC-N1003 mostrada anteriormente, si candIntraPredModeA no es igual a candIntraPredModeB, el codificador de vídeo puede incluir en la lista de candidatos a MPM otros modos de intrapredicción angulares determinados como los más aproximados a candIntraPredModeA o candIntraPredModeB usando las fórmulas mostradas en la versión adaptada de ejemplo de la subcláusula 8.4.2 de JCTVC-N1003 que incluye operaciones de módulo.

[0203] Como se describe anteriormente, en algunos ejemplos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar más de 33 modos de intrapredicción angulares. El siguiente texto describe ejemplos de cambios en JCTVC-N1003 para implementar 65 modos de intrapredicción angulares.

8.4.4.2.6 Especificación del modo de intrapredicción en el intervalo de INTRA_ANGULAR2.. INTRA_ANGULAR34

[0204] Las entradas para este proceso son:

- el modo de intrapredicción predModeIntra,
- las muestras vecinas p[x][y], con $x = -1$, $y = -1 \dots nTbS * 2 - 1$ y $x = 0 \dots nTbS * 2 - 1$, $y = -1$,
- una variable nTbS que especifica el tamaño de bloque de transformada,
- una variable cldx que especifica la componente de color del bloque actual.

[0205] Las salidas de este proceso son las muestras predichas predSamples[x][y], con $x, y = 0 \dots nTbS - 1$.

[0206] La FIG. 8-2, que es la FIG. 9 de esta divulgación, ilustra el total de 33 ángulos de intrapredicción y la Tabla 8-4 especifica la tabla de correlación entre predModeIntra y el parámetro de ángulo intraPredAngle.

Tabla 8-4 - Especificación de intraPredAngle

predMode Intra	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
intraPred Angle	-	32	26	21	17	13	9	5	2	0	-2	-5	-9	-13	-17	-21	-26
predMode Intra	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
intraPred Angle	-32	-26	-21	-17	-13	-9	-5	-2	0	2	5	9	13	17	21	26	32

Tabla 8-4 - Especificación de intraPredAngle

predMode Intra	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
intraPred Angle	-	-	32	29	26	23	21	19	17	15	13	11	9	7	5	3	2

predMode Intra	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
intraPred Angle	1	0	1	2	3	5	7	9	11	13	15	17	19	21	23	26	29
predMode Intra	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
intraPred Angle	32	29	26	23	21	19	17	15	13	11	9	7	5	3	2	1	0
predMode Intra	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67
intraPred Angle	1	2	3	5	7	9	11	13	15	17	19	21	23	26	29	32	-

</ins>

[0207] La Tabla 8-5 especifica además la tabla de correlación entre predModelIntra y el parámetro de ángulo inverso invAngle.

Tabla 8-5 - Especificación de invAngle

predModelIntra	11	12	13	14	15	16	17	18
invAngle	-4096	-1638	-910	-630	-482	-390	-315	-256
predModelIntra	19	20	21	22	23	24	25	26
invAngle	-315	-390	-482	-630	-910	-1638	-4096	-

[0208] Los valores de las muestras de predicción predSamples[x][y], con $x, y = 0..nTbS - 1$, son obtenidos como sigue:

- Si predModelIntra es igual a o mayor que 18, se aplican las siguientes etapas ordenadas:

1. La matriz de muestra de referencia ref[x] se especifica como sigue:

- Se aplica lo siguiente:

$ref[x] = p[-1 + x][-1]$, con $x = 0..nTbS$ (8-47)

- Si intraPredAngle es menor que 0, la matriz de muestras de referencia principal se amplía como sigue:

- Cuando $(nTbS * intraPredAngle) \gg 5$ es menor que -1,

$ref[x] = p[-1][-1 + ((x * invAngle + 128) \gg 8)]$, con $x = -1..(nTbS * intraPredAngle) \gg 5$ (8-48)

- De lo contrario,

$ref[x] = p[-1 + x][-1]$, con $x = nTbS + 1..2 * nTbS$ (8-49)

2. Los valores de las muestras de predicción predSamples[x][y], con $x, y = 0..nTbS - 1$, son obtenidos como sigue:

a. La variable de índice ildx y el factor de multiplicación iFact se obtienen como sigue:

$ildx = ((y + 1) * intraPredAngle) \gg 5$ (8-50)

$iFact = ((y + 1) * intraPredAngle) \& 31$ (8-51)

b. Dependiendo del valor de iFact, se aplica lo siguiente:

- Si iFact no es igual a 0, el valor de las muestras de predicción predSamples[x][y] se obtiene como sigue:

$predSamples[x][y] = ((32 - iFact) * ref[x + ildx + 1] + iFact * ref[x + ildx + 2] + 16) \gg 5$ (8-52)

- De lo contrario, el valor de las muestras de predicción predSamples[x][y] se obtiene como sigue:

$predSamples[x][y] = ref[x + ildx + 1]$ (8-53)

c. Cuando predModelIntra es igual a 26 (vertical), cldx es igual a 0 y nTbS es menor que 32, se aplica el siguiente filtrado con $x = 0$, $y = 0..\text{nTbS} - 1$:

$$\text{predSamples}[x][y] = \text{Clip1}_v(p[x][-1] + ((p[-1][y] - p[-1][-1]) >> 1)) \quad (8-54)$$

- De lo contrario (predModelIntra es menor que 18), se aplican las siguientes etapas ordenadas:

1. La matriz de muestra de referencia $\text{ref}[x]$ se especifica como sigue:

- Se aplica lo siguiente:

$$\text{ref}[x] = p[-1][-1+x], \text{ con } x = 0..\text{nTbS} \quad (8-55)$$

- Si intraPredAngle es menor que 0, la matriz de muestras de referencia principal se amplía como sigue:

- Cuando $(\text{nTbS} * \text{intraPredAngle}) >> 5$ es menor que -1,

$$\text{ref}[x] = p[-1 + ((x * \text{invAngle} + 128) >> 8)][-1], \text{ con } x = -1..(\text{nTbS} * \text{intraPredAngle}) >> 5 \quad (8-56)$$

- De lo contrario,

$$\text{ref}[x] = p[-1][-1+x], \text{ con } x = \text{nTbS} + 1..2 * \text{nTbS} \quad (8-57)$$

2. Los valores de las muestras de predicción $\text{predSamples}[x][y]$, con $x, y = 0..\text{nTbS} - 1$, son obtenidos como sigue:

a. La variable de índice ildx y el factor de multiplicación iFact se obtienen como sigue:

$$\text{ildx} = ((x + 1) * \text{intraPredAngle}) >> 5 \quad (8-58)$$

$$\text{iFact} = ((x + 1) * \text{intraPredAngle}) \& 31 \quad (8-59)$$

b. Dependiendo del valor de iFact , se aplica lo siguiente:

- Si iFact no es igual a 0, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

$$\text{predSamples}[x][y] = ((32 - \text{iFact}) * \text{ref}[y + \text{ildx} + 1] + \text{iFact} * \text{ref}[y + \text{ildx} + 2] + 16) >> 5 \quad (8-60)$$

- De lo contrario, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

$$\text{predSamples}[x][y] = \text{ref}[y + \text{ildx} + 1] \quad (8-61)$$

c. Cuando predModelIntra es igual a 10 (horizontal), cldx es igual a 0 y nTbS es menor que 32, se aplica el siguiente filtrado con $x = 0..\text{nTbS} - 1$, $y = 0$:

$$\text{predSamples}[x][y] = \text{Clip1}_v(p[-1][y] + ((p[x][-1] - p[-1][-1]) >> 1)) \quad (8-62)$$

[0209] La Tabla 8-5 especifica además la tabla de correlación entre predModelIntra y el parámetro de ángulo inverso invAngle .

Tabla 8-5 - Especificación de invAngle

predModelIntra	11	12	13	14	15	16	17	18
invAngle	-4096	-1638	-910	-630	-482	-390	-315	-256
predModelIntra	19	20	21	22	23	24	25	26
invAngle	-315	-390	-482	-630	-910	-1638	-4096	-

[0210] Los valores de las muestras de predicción $\text{predSamples}[x][y]$, con $x, y = 0..\text{nTbS} - 1$, son obtenidos como sigue:

- Si predModelIntra es igual a o mayor que 18, se aplican las siguientes etapas ordenadas:

7.4.9.11 Semántica de codificación residual

[0211] En la intrapredicción se usan diferentes órdenes de exploración. La variable scanIdx especifica qué orden de exploración se usa, donde scanIdx igual a 0 especifica un orden de exploración diagonal superior- derecha, scanIdx igual a 1 especifica un orden de exploración horizontal y scanIdx igual a 2 especifica un orden de exploración vertical. El valor de scanIdx se obtiene como sigue:

- Si CuPredMode[x0][y0] es igual a MODE_INTRA y una o más de las siguientes condiciones son verdaderas:

- log2TrafoSize es igual a 2.

- log2TrafoSize es igual a 3 y cldx es igual a 0.

predModelIntra se obtiene como sigue:

- Si cldx es igual a 0, predModelIntra se establece igual a IntraPredModeY[x0][y0].

- De lo contrario, predModelIntra se establece igual a IntraPredModeC.

scanIdx se obtiene como sigue:

- Si predModelIntra está en el intervalo de $\langle \text{dlt} \rangle 6 \langle \text{dlt} \rangle \langle \text{ins} \rangle 10 \langle \text{ins} \rangle$ a $\langle \text{dlt} \rangle 14 \langle \text{dlt} \rangle \langle \text{ins} \rangle 26 \langle \text{ins} \rangle$, ambos inclusive, scanIdx se establece igual a 2.

- De lo contrario, si predModelIntra está en el intervalo de $\langle \text{dlt} \rangle 22 \langle \text{dlt} \rangle \langle \text{ins} \rangle 42 \langle \text{ins} \rangle$ a $\langle \text{dlt} \rangle 30 \langle \text{dlt} \rangle \langle \text{ins} \rangle 58 \langle \text{ins} \rangle$, ambos inclusive, scanIdx se establece igual a 1.

- De lo contrario, scanIdx se establece igual a 0.

- De lo contrario, scanIdx se establece igual a 0.

8.4.4.2.3 Proceso de filtrado de muestras vecinas

[0212] Las entradas para este proceso son:

- las muestras vecinas p[x][y], con $x = -1$, $y = -1 \dots nTbS * 2 - 1$ y $x = 0 \dots nTbS * 2 - 1$, $y = -1$,

- una variable nTbS que especifica el tamaño de bloque de transformada.

[0213] Las salidas de este proceso son las muestras filtradas pF[x][y], con

$x = -1$, $y = -1 \dots nTbS * 2 - 1$ y $x = 0 \dots nTbS * 2 - 1$, $y = -1$.

[0214] La variable filterFlag se obtiene como sigue:

- Si una o más de las siguientes condiciones son verdaderas, filterFlag se establece igual a 0:

- predModelIntra es igual a INTRA_DC.

- nTbS es igual 4.

- De lo contrario, se aplica lo siguiente:

- La variable minDistVerHor se establece igual a $\text{Min}(\text{Abs}(\text{predModelIntra} - 26), \text{Abs}(\text{predModelIntra} - 10))$.

- La variable intraHorVerDistThres[nTbS] se especifica en la Tabla 8-3.

- La variable filterFlag se obtiene como sigue:

- Si minDistVerHor es mayor que intraHorVerDistThres[nTbS], filterFlag se establece igual a 1.

- De lo contrario, filterFlag se establece igual a 0.

7.4.9.11 Semántica de codificación residual

[0215] En la intrapredicción se usan diferentes órdenes de exploración. La variable scanIdx especifica qué orden de exploración se usa, donde scanIdx igual a 0 especifica un orden de exploración diagonal superior- derecha, scanIdx

igual a 1 especifica un orden de exploración horizontal y scanIdx igual a 2 especifica un orden de exploración vertical. El valor de scanIdx se obtiene como sigue:

- Si CuPredMode[x0][y0] es igual a MODE_INTRA y una o más de las siguientes condiciones son verdaderas:

- log2TrafoSize es igual a 2.
- log2TrafoSize es igual a 3 y cldx es igual a 0.

predModelIntra se obtiene como sigue:

- Si cldx es igual a 0, predModelIntra se establece igual a IntraPredModeY[x0][y0].
- De lo contrario, predModelIntra se establece igual a IntraPredModeC.

scanIdx se obtiene como sigue:

- Si predModelIntra está en el intervalo de $\langle \text{dlt} > 6 \text{ } \langle \text{ins} > 10 \text{ } \langle \text{ins} > 14 \text{ } \langle \text{ins} > 26 \text{ } \langle \text{ins} >$, ambos inclusive, scanIdx se establece igual a 2.
- De lo contrario, si predModelIntra está en el intervalo de $\langle \text{dlt} > 22 \text{ } \langle \text{ins} > 42 \text{ } \langle \text{ins} > 30 \text{ } \langle \text{ins} > 58 \text{ } \langle \text{ins} >$, ambos inclusive, scanIdx se establece igual a 1.
- De lo contrario, scanIdx se establece igual a 0.

- De lo contrario, scanIdx se establece igual a 0.

8.4.4.2.3 Proceso de filtrado de muestras vecinas

[0216] Las entradas para este proceso son:

- las muestras vecinas p[x][y], con $x = -1$, $y = -1..nTbS * 2 - 1$ y $x = 0..nTbS * 2 - 1$, $y = -1$,
- una variable nTbS que especifica el tamaño de bloque de transformada.

[0217] Las salidas de este proceso son las muestras filtradas pF[x][y], con $x = -1$, $y = -1..nTbS * 2 - 1$ y $x = 0..nTbS * 2 - 1$, $y = -1$.

[0218] La variable filterFlag se obtiene como sigue:

- Si una o más de las siguientes condiciones son verdaderas, filterFlag se establece igual a 0:
 - predModelIntra es igual a INTRA_DC.
 - nTbS es igual 4.
- De lo contrario, se aplica lo siguiente:
 - La variable minDistVerHor se establece igual a $\text{Min}(\text{Abs}(\text{predModelIntra} - 26), \text{Abs}(\text{predModelIntra} - 10))$.
 - La variable intraHorVerDistThres[nTbS] se especifica en la Tabla 8-3.
 - La variable filterFlag se obtiene como sigue:
 - Si minDistVerHor es mayor que intraHorVerDistThres[nTbS], filterFlag se establece igual a 1.
 - De lo contrario, filterFlag se establece igual a 0.

Tabla 8-3 - Especificación de intraHorVerDistThres[nTbS] para varios tamaños de bloque de transformada

	nTbS = 8	nTbS = 16	nTbS = 32
intraHorVerDistThres [nTbS]	$\langle \text{dlt} > 7 \text{ } \langle \text{ins} > 14 \text{ } \langle \text{ins} >$	$\langle \text{dlt} > 1 \text{ } \langle \text{ins} > 2 \text{ } \langle \text{ins} >$	0

[0219] Cuando filterFlag es igual a 1, se aplica lo siguiente:

[0220] Como se describe anteriormente, de acuerdo con algunas técnicas de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden aplicar un filtro de interpolación intra de N etapas. El siguiente texto describe ejemplos de cambios en JCTVC-N1003 para implementar la aplicación de un filtro de interpolación intra de 4 etapas.

<ins>x.x.x Proceso de inicialización de coeficientes de filtro de interpolación intra</ins>

[0221] La salida de este proceso es la matriz filterCubic[sFrac][pos]. El índice de matriz sFrac especifica la posición fraccionaria que va de 0 a 31, pos especifica el coeficiente de filtro para la muestra pos-ésima. La matriz filterCubic y filterGaussian se obtiene como sigue:

```

<ins>filterCubic [32][4] = {
{
{ 0, 256, 0, 0 }, // 0
{ -3, 252, 8, -1 }, // 1
{ -5, 247, 17, -3 }, // 2
{ -7, 242, 25, -4 }, // 3
{ -9, 236, 34, -5 }, // 4
{ -10, 230, 43, -7 }, // 5
{ -12, 224, 52, -8 }, // 6
{ -13, 217, 61, -9 }, // 7
{ -14, 210, 70, -10 }, // 8
{ -15, 203, 79, -11 }, // 9
{ -16, 195, 89, -12 }, // 10
{ -16, 187, 98, -13 }, // 11
{ -16, 179, 107, -14 }, // 12
{ -16, 170, 116, -14 }, // 13
{ -17, 162, 126, -15 }, // 14
{ -16, 153, 135, -16 }, // 15
{ -16, 144, 144, -16 }, // 16
},
};

sigma = 0,9
for(i=0; i<17; i++)
{

```

```

    for(c=0; c<4; c++)
    {
5        filterGaussian [b][i][c] = (256,0 * exp(-((c-delta)/sigma)2) / sum + 0,5);
    }
10    }
    for(i=17; i<32; i++)
    {
15        for(c=0; c<4; c++)
        {
            filterCubic [b][i][c] = filterCubic [b][32-i][3-c];
20            filterGaussian [b][i][c] = filterGaussian [b][32-i][3-c];
        }
25    }
</ins>

```

8.4.4.2.6 Especificación del modo de intrapredicción en el intervalo de INTRA ANGULAR2.. INTRA ANGULAR34

[0222] Las entradas para este proceso son:

- el modo de intrapredicción predModelIntra,
- las muestras vecinas p[x][y], con $x = -1$, $y = -1..nTbS * 2 - 1$ y $x = 0..nTbS * 2 - 1$, $y = -1$,
- una variable nTbS que especifica el tamaño de bloque de transformada,
- una variable cldx que especifica la componente de color del bloque actual.

[0223] Las salidas de este proceso son las muestras predichas predSamples[x][y], con $x, y = 0..nTbS - 1$.

[0224] Los valores de las muestras de predicción predSamples[x][y], con $x, y = 0..nTbS - 1$, son obtenidos como sigue:

- Si predModelIntra es igual a o mayor que 18, se aplican las siguientes etapas ordenadas:

3. La matriz de muestra de referencia ref[x] se especifica como sigue:

- Se aplica lo siguiente:

$ref[x] = p[-1 + x][-1]$, con $x = 0..nTbS$ (8-47)

- Si intraPredAngle es menor que 0, la matriz de muestras de referencia principal se amplía como sigue:

- Cuando $(nTbS * intraPredAngle) >> 5$ es menor que -1,

$ref[x] = p[-1][-1 + ((x * invAngle + 128) >> 8)]$, con $x = -1..(nTbS * intraPredAngle) >> 5$ (8-48)

- De lo contrario,

$ref[x] = p[-1 + x][-1]$, con $x = nTbS + 1..2 * nTbS$ (8-49)

4. Los valores de las muestras de predicción $\text{predSamples}[x][y]$, con $x, y = 0..n\text{TbS} - 1$, son obtenidos como sigue:

a. La variable de índice ildx y el factor de multiplicación iFact se obtienen como sigue:

$\text{ildx} = ((y + 1) * \text{intraPredAngle}) \gg 5$ (8-50)

$\text{iFact} = ((y + 1) * \text{intraPredAngle}) \& 31$ (8-51)

b. Dependiendo del valor de iFact , se aplica lo siguiente:

- Si iFact no es igual a 0, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

- $\text{predSamples}[x][y] = ((32 - \text{iFact}) * \text{ref}[x + \text{ildx} + 1] + \text{iFact} * \text{ref}[x + \text{ildx} + 2] + 16) \gg 5$ (8-52)
 For $p=0,...,3$, $\text{pF}[p] = (\text{cldx}=0 \&\& n\text{TbS} \leq 8) ? \text{filterCubic}[\text{iFact}][p] : \text{filterGaussian}[\text{iFact}][p]$
 $\text{P}[1] = \text{ref}[x + \text{ildx} + 1]$ $\text{P}[2] = \text{ref}[x + \text{ildx} + 2]$ $\text{P}[0] = (x=0) ? \text{ref}[x + \text{ildx} + 1] : \text{ref}[x + \text{ildx}]$ $\text{P}[3] = (x=n\text{TbS}-1) ? \text{ref}[x + \text{ildx} + 2] : \text{ref}[x + \text{ildx} + 3]$
 $\text{predSamples}[x][y] = (\text{pF}[0] * \text{P}[0] + \text{pF}[1] * \text{P}[1] + \text{pF}[2] * \text{P}[2] + \text{pF}[3] * \text{P}[3] + 128) \gg 8$ (8-52)

- De lo contrario, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

- $\text{predSamples}[x][y] = \text{ref}[x + \text{ildx} + 1]$ (8-53)

c. Cuando predModelIntra es igual a 26 (vertical), cldx es igual a 0 y $n\text{TbS}$ es menor que 32, se aplica el siguiente filtrado con $x = 0, y = 0..n\text{TbS} - 1$:

$\text{predSamples}[x][y] = \text{Clip1}(\text{p}[x][-1] + ((\text{p}[-1][y] - \text{p}[-1][-1]) \gg 1))$ (8-54)

- De lo contrario (predModelIntra es menor que 18), se aplican las siguientes etapas ordenadas:

3. La matriz de muestra de referencia $\text{ref}[x]$ se especifica como sigue:

- Se aplica lo siguiente:

$\text{ref}[x] = \text{p}[-1][-1+x]$, con $x = 0..n\text{TbS}$ (8-55)

- Si intraPredAngle es menor que 0, la matriz de muestras de referencia principal se amplía como sigue:

- Cuando $(n\text{TbS} * \text{intraPredAngle}) \gg 5$ es menor que -1,

$\text{ref}[x] = \text{p}[-1 + ((x * \text{invAngle} + 128) \gg 8)][-1]$, con $x = -1..(n\text{TbS} * \text{intraPredAngle}) \gg 5$ (8-56)

- De lo contrario,

$\text{ref}[x] = \text{p}[-1][-1+x]$, con $x = n\text{TbS} + 1..2 * n\text{TbS}$ (8-57)

4. Los valores de las muestras de predicción $\text{predSamples}[x][y]$, con $x, y = 0..n\text{TbS} - 1$, son obtenidos como sigue:

d. La variable de índice ildx y el factor de multiplicación iFact se obtienen como sigue:

$\text{ildx} = ((x+1) * \text{intraPredAngle}) \gg 5$ (8-58) $\text{iFact} = ((x+1) * \text{intraPredAngle}) \& 31$ (8-59)

e. Dependiendo del valor de iFact , se aplica lo siguiente:

- Si iFact no es igual a 0, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

$\text{predSamples}[x][y] = ((32 - \text{iFact}) * \text{ref}[y + \text{ildx} + 1] + \text{iFact} * \text{ref}[y + \text{ildx} + 2] + 16) \gg 5$ (8-60)

- De lo contrario, el valor de las muestras de predicción $\text{predSamples}[x][y]$ se obtiene como sigue:

$\text{predSamples}[x][y] = \text{ref}[y + \text{ildx} + 1]$ (8-61)

f. Cuando predModelIntra es igual a 10 (horizontal), cldx es igual a 0 y nTbS es menor que 32, se aplica el siguiente filtrado con $x = 0..nTbS - 1$, $y = 0$:

$$\text{predSamples}[x][y] = \text{Clip1}_v(p[-1][y] + ((p[x][-1] - p[-1][-1]) >> 1)) \quad (8-62)$$

[0225] La FIG. 10 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 10 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo obtiene, de entre una pluralidad de modos de intrapredicción, M MPM para la intrapredicción de un bloque de datos de vídeo (200). En un ejemplo, M puede ser mayor que 3. El codificador de vídeo codifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado (202) de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en que el índice indicado de entre el índice MPM o el índice no MPM sea el índice MPM, el codificador de vídeo selecciona, para cada *bin* del uno o más *bins* modelados por contexto del índice MPM, en base a modos de intrapredicción usados para descodificar uno o más bloques vecinos, un índice de contexto para el *bin* modelado por contexto (204). El codificador de vídeo codifica el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (206).

[0226] La FIG. 11 es un diagrama de flujo que ilustra un procedimiento de descodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 11 se describe con referencia a un descodificador de vídeo genérico. El descodificador de vídeo genérico puede, por ejemplo, corresponder al descodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de descodificador de vídeo. El descodificador de vídeo obtiene, de entre una pluralidad de modos de intrapredicción, M MPM para la intrapredicción de un bloque de datos de vídeo (220). En un ejemplo, M puede ser mayor que 3. El descodificador de vídeo codifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado (222) de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo. El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en que el índice indicado de entre el índice MPM o el índice no MPM sea el índice MPM, el descodificador de vídeo selecciona, para cada *bin* del uno o más *bins* modelados por contexto del índice MPM, en base a modos de intrapredicción usados para descodificar uno o más bloques vecinos, un índice de contexto para el *bin* modelado por contexto (224). El descodificador de vídeo reconstruye el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (226).

[0227] La FIG. 12 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 12 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo obtiene M MPM (300) para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. En un ejemplo, M puede ser mayor que 3 y los MPM pueden incluir un MPM para una columna vecina izquierda y un MPM para una fila vecina superior. En un ejemplo, los M MPM pueden obtenerse mediante al menos de uno de (i) definir un modo de intrapredicción representativo para la columna vecina izquierda y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda (302), o (ii) definir un modo de intrapredicción representativo para la fila vecina superior y usar el modo de intrapredicción representativo para la fila vecina superior como el MPM para la fila vecina superior (304).

[0228] El codificador de vídeo puede descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (306). Además, el codificador de vídeo puede codificar el índice indicado del índice MPM o el índice no MPM (308). El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. El codificador de vídeo puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (310).

[0229] La FIG. 13 es un diagrama de flujo que ilustra un procedimiento de descodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 13 se describe con referencia a un descodificador de vídeo genérico. El descodificador de vídeo genérico puede, por ejemplo, corresponder al descodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de descodificador de vídeo. El descodificador de vídeo obtiene M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (350). En un ejemplo, M puede ser mayor que 3 y los MPM pueden incluir un MPM para una columna vecina izquierda y un MPM para una fila vecina superior. Los M modos más probables pueden obtenerse mediante al menos de uno de (i) definir un modo de intrapredicción representativo para la columna vecina izquierda y usar el modo de intrapredicción representativo para la columna vecina izquierda como el MPM para la columna vecina izquierda

(352), o (ii) definir un modo de intrapredicción representativo para la fila vecina superior y usar el modo de intrapredicción representativo para la fila vecina superior como el MPM para la fila vecina superior (354).

[0230] El decodificador de vídeo puede decodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (356). Además, el decodificador de vídeo puede decodificar el índice indicado del índice MPM o el índice no MPM (358). El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. El decodificador de vídeo puede codificar el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (360).

[0231] La FIG. 14 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 14 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo obtiene M MPM (400) para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción. En un ejemplo, M puede ser mayor que 3. El codificador de vídeo codifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (402). En el ejemplo de la FIG. 12, el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. Además, el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en el índice MPM que indica el modo de intrapredicción seleccionado, el codificador de vídeo codifica el índice no MPM (404). En el ejemplo de la FIG. 14, el índice no MPM se codifica en el flujo de bits como una palabra de código más corta que $\lceil \log_2 N \rceil$ bits si el índice no MPM cumple un criterio y, de lo contrario, se codifica en el flujo de bits como un código de longitud fija con $\lceil \log_2 N \rceil$ bits. En el ejemplo de la FIG. 14, hay un total de N valores disponibles del índice no MPM. El codificador de vídeo codifica el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (406).

[0232] La FIG. 15 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 15 se describe con referencia a un decodificador de vídeo genérico. El decodificador de vídeo genérico puede, por ejemplo, corresponder al decodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de decodificador de vídeo. El decodificador de vídeo obtiene M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (450). En un ejemplo, M puede ser mayor que 3. El decodificador de vídeo decodifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (452). El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. Basándose en el índice MPM que indica el modo de intrapredicción seleccionado, el decodificador de vídeo decodifica el índice no MPM (454). En el ejemplo de la FIG. 15, el índice no MPM se codifica en el flujo de bits como una palabra de código más corta que $\lceil \log_2 N \rceil$ bits si el índice no MPM cumple un criterio y, de lo contrario, se codifica en el flujo de bits como un código de longitud fija con $\lceil \log_2 N \rceil$ bits. En el ejemplo de la FIG. 15, puede haber un total de N valores disponibles del índice no MPM. Además, en el ejemplo de la FIG. 15, el decodificador de vídeo reconstruye el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (456).

[0233] La FIG. 16 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 16 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo obtiene M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (600). En el ejemplo de la FIG. 16, M es mayor que 3. El codificador de vídeo codifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (602). El codificador de vídeo codifica el índice indicado del índice MPM o el índice no MPM (604). El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado. El codificador de vídeo codifica el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (606).

[0234] La FIG. 17 es un diagrama de flujo que ilustra un procedimiento de decodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 17 se describe con referencia a un decodificador de vídeo genérico. El decodificador de vídeo genérico puede, por ejemplo, corresponder al decodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de decodificador de vídeo. El decodificador de vídeo obtiene M MPM para la intrapredicción del bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (650). En el ejemplo de la FIG. 17, M es mayor que 3. El decodificador de vídeo decodifica un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque de datos de vídeo (652).

El descodificador de vídeo descodifica el índice indicado del índice MPM o el índice no MPM (654). El índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado. El índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado (654). El descodificador de vídeo reconstruye el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (656).

[0235] La FIG. 18 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 18 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo codifica información sintáctica que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (700). En un ejemplo, la pluralidad de modos de intrapredicción puede incluir más de 33 modos de intrapredicción angulares. Los modos de intrapredicción angulares pueden definirse de modo que la interpolación se realice con una precisión de 1/32 pel. El codificador de vídeo codifica el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (702).

[0236] La FIG. 19 es un diagrama de flujo que ilustra un procedimiento de descodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 19 se describe con referencia a un descodificador de vídeo genérico. El descodificador de vídeo genérico puede, por ejemplo, corresponder al descodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de descodificador de vídeo. El descodificador de vídeo descodifica información sintáctica que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (750). En un ejemplo, la pluralidad de modos de intrapredicción puede incluir más de 33 modos de intrapredicción angulares, y los modos de intrapredicción angulares pueden definirse de modo que la interpolación se realice con una precisión de 1/32 pel. El descodificador de vídeo puede reconstruir el bloque de datos de vídeo basándose en el modo de intrapredicción seleccionado (752).

[0237] La FIG. 20 es un diagrama de flujo que ilustra un procedimiento de codificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 20 se describe con referencia a un codificador de vídeo genérico. El codificador de vídeo genérico puede, por ejemplo, corresponder al codificador de vídeo 20, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de codificador de vídeo. El codificador de vídeo codifica información sintáctica que indica un modo de intrapredicción seleccionado para el bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (800). El codificador de vídeo aplica un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas del bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado (802). En un ejemplo, N puede ser mayor que 2. El codificador de vídeo puede codificar el bloque de datos de vídeo en base a las muestras reconstruidas vecinas filtradas de acuerdo con el modo de intrapredicción seleccionado (804).

[0238] La FIG. 21 es un diagrama de flujo que ilustra un procedimiento de descodificación de datos de vídeo de acuerdo con técnicas de esta divulgación. La FIG. 21 se describe con referencia a un descodificador de vídeo genérico. El descodificador de vídeo genérico puede, por ejemplo, corresponder al descodificador de vídeo 30, aunque las técnicas de esta divulgación no se limitan a ningún tipo específico de descodificador de vídeo. El descodificador de vídeo descodifica información sintáctica que indica un modo de intrapredicción seleccionado para un bloque de datos de vídeo de entre una pluralidad de modos de intrapredicción (850). El descodificador de vídeo aplica un filtro de interpolación intra de N etapas a muestras reconstruidas vecinas del bloque de datos de vídeo de acuerdo con el modo de intrapredicción seleccionado (852). En un ejemplo, N puede ser mayor que 2. El descodificador de vídeo puede reconstruir el bloque de datos de vídeo en base a las muestras reconstruidas vecinas filtradas de acuerdo con el modo de intrapredicción seleccionado (854).

[0239] En uno o más ejemplos, las funciones descritas se pueden implementar en hardware, software, firmware o en cualquier combinación de los mismos. Si se implementan en software, las funciones se pueden almacenar en, o transmitir por, un medio legible por ordenador como una o más instrucciones o código, y ejecutar mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como medios de almacenamiento de datos, o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, en general, a (1) medios de almacenamiento tangibles legibles por ordenador que son no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder mediante uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0240] A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que se pueda usar para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que se pueda

acceder mediante un ordenador. Además, cualquier conexión recibe apropiadamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o tecnologías inalámbricas tales como infrarrojos, radio y microondas están incluidos en la definición de medio. Sin embargo, se deberá entender que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, están dirigidos a medios de almacenamiento no transitorio tangibles. Los discos, como se usan en el presente documento, incluyen el disco compacto (CD), el disco láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos reproducen normalmente los datos de forma magnética, mientras que otros discos reproducen los datos de forma óptica con láseres. Las combinaciones de lo anterior también se deben incluir dentro del alcance de los medios legibles por ordenador.

[0241] Uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), matrices lógicas programables *in situ* (FPGA) u otros circuitos lógicos integrados o discretos equivalentes pueden ejecutar instrucciones. En consecuencia, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar en módulos de hardware y/o de software dedicados configurados para la codificación y la decodificación, o incorporar en un códec combinado. Además, las técnicas se podrían implementar por completo en uno o más circuitos o elementos lógicos.

[0242] Las técnicas de esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). En esta divulgación se describen diversos componentes, módulos o unidades para destacar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se describe anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec o proporcionar mediante un grupo de unidades de hardware interoperativas, que incluyen uno o más procesadores descritos anteriormente, junto con software y/o firmware adecuados.

[0243] Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento de descodificación de un bloque actual de datos de vídeo, comprendiendo el procedimiento:

- 5 obtener (350) M modos más probables, MPM, para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y la obtención de los M modos más probables comprende al menos uno de:
 - 10 (i) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda;

definir (352) un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y
15 usar el modo de intrapredicción representativo definido para la columna vecina izquierda como el MPM para la columna vecina izquierda, o
 - 20 (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior;

definir (354) un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y
25 usar el modo de intrapredicción representativo definido para la fila vecina superior como el MPM para la fila vecina superior;
- 30 descodificar (356) un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo;
- 35 descodificar (358) el índice indicado del índice de modo más probable o el índice de modo que no es más probable, donde el índice de modo más probable indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y
- reconstruir (360) el bloque actual de datos de vídeo en función del modo de intrapredicción seleccionado.

40 2. El procedimiento de la reivindicación 1, en el que:

- el bloque actual de datos de vídeo es una unidad de codificación, CU, actual en una imagen actual de los datos de vídeo,
- 45 el modo de intrapredicción seleccionado para el bloque actual de datos de vídeo es un modo de intrapredicción seleccionado para una unidad de predicción, PU, actual de la CU actual, y la reconstrucción (358) del bloque actual de datos de vídeo comprende:

generar un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado; y
50 reconstruir la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de unidades de transformada, TU, de la CU actual.

55 3. Un procedimiento de codificación de un bloque actual de datos de vídeo, comprendiendo el procedimiento:

- 60 obtener (300) M modos más probables, MPM, para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y la obtención de los M modos más probables comprende al menos uno de:
 - (i) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda;

definir (302) un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y

usar el modo de intrapredicción representativo definido para la columna vecina izquierda como el MPM para la columna vecina izquierda, o

(ii) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior;

definir (304) un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y

usar el modo de intrapredicción representativo definido para la fila vecina superior como el MPM para la fila vecina superior;

codificar (306) un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo;

codificar (308) el índice indicado del índice MPM o el índice no MPM, donde el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y

codificar (310) el bloque actual de datos de vídeo en función del modo de intrapredicción seleccionado.

4. El procedimiento de la reivindicación 3, en el que:

el bloque actual de datos de vídeo es una unidad de codificación, CU, actual en una imagen actual de los datos de vídeo,

el modo de intrapredicción seleccionado para el bloque actual de datos de vídeo es un modo de intrapredicción seleccionado para una unidad de predicción, PU, actual de la CU actual, y la codificación (310) del bloque actual de datos de vídeo en base al modo de intrapredicción seleccionado comprende:

generar, en base al modo de intrapredicción seleccionado de la PU, un bloque predictivo para la PU; y

generar datos residuales que representan diferencias de píxel entre la CU actual y el bloque predictivo.

5. Un dispositivo (30) para descodificar un bloque actual de datos de vídeo, comprendiendo el dispositivo:

una memoria configurada para almacenar los datos de vídeo; y

uno o más procesadores configurados para:

obtener M modos más probables, MPM, para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y el uno o más procesadores están configurados de modo que, como parte de la obtención de los M modos más probables, el uno o más procesadores realizan al menos uno de:

(i) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda;

definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y

usar el modo de intrapredicción representativo definido para la columna vecina izquierda como el MPM para la columna vecina izquierda, o

(ii) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior;

definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y

5 usar el modo de intrapredicción representativo definido para la fila vecina superior como el MPM para la fila vecina superior;

10 descodificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo;

15 descodificar el índice indicado del índice de modo más probable o el índice de modo que no es más probable, donde el índice de modo más probable indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y

reconstruir el bloque actual de datos de vídeo en función del modo de intrapredicción seleccionado.

20 6. El dispositivo (30) de la reivindicación 5, en el que:

el bloque actual de datos de vídeo es una unidad de codificación, CU, actual en una imagen actual de los datos de vídeo,

25 el modo de intrapredicción seleccionado para el bloque actual de datos de vídeo es un modo de intrapredicción seleccionado para una unidad de predicción, PU, actual de la CU actual, y

los uno o más procesadores están configurados de modo que, como parte de la reconstrucción del bloque actual de datos de vídeo, los uno o más procesadores:

30 generan un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado; y

reconstruyen la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de unidades de transformada, TU, de la CU actual.

35 7. Un dispositivo (20) para codificar un bloque actual de datos de vídeo, comprendiendo el dispositivo:

una memoria configurada para almacenar los datos de vídeo; y

40 uno o más procesadores configurados para:

45 obtener M modos más probables, MPM, para la intrapredicción del bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y el uno o más procesadores están configurados de modo que, como parte de la obtención de los M modos más probables, el uno o más procesadores realizan al menos uno de:

50 (i) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda;

definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y

55 usar el modo de intrapredicción representativo definido para la columna vecina izquierda como el MPM para la columna vecina izquierda, o

60 (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior;

definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y

65 usar el modo de intrapredicción representativo definido para la fila vecina superior como el MPM para la fila vecina superior;

codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción del bloque actual de datos de vídeo;

5 codificar el índice indicado del índice de modo más probable o el índice de modo que no es más probable, donde el índice de modo más probable indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y

10 codificar el bloque actual de datos de vídeo en función del modo de intrapredicción seleccionado.

8. El dispositivo (20) de la reivindicación 7, en el que:

15 el bloque actual de datos de vídeo es una unidad de codificación, CU, actual en una imagen actual de los datos de vídeo,

el modo de intrapredicción seleccionado para el bloque actual de datos de vídeo es un modo de intrapredicción seleccionado para una unidad de predicción, PU, actual de la CU actual, y

20 los uno o más procesadores están configurados de modo que, como parte de la reconstrucción del bloque actual de datos de vídeo, los uno o más procesadores:

generan un bloque predictivo para la PU actual usando el modo de intrapredicción seleccionado; y

25 reconstruyen la CU actual usando valores residuales añadiendo muestras de los bloques predictivos de las PU de la CU a las muestras correspondientes de bloques de transformada de unidades de transformada, TU, de la CU actual.

9. Un medio legible por ordenador que almacena instrucciones que, cuando son ejecutadas por uno o más procesadores hacen que los uno o más procesadores:

30 obtengan M modos más probables, MPM, para la intrapredicción de un bloque actual de datos de vídeo de entre una pluralidad de modos de intrapredicción, donde M es mayor que 3, incluyendo los MPM un MPM para una columna vecina izquierda y un MPM para una fila vecina superior, y como parte de hacer que el uno o más procesadores obtengan los M modos más probables, las instrucciones, cuando son ejecutadas por el uno o más procesadores, hacen que los uno o más procesadores realicen al menos uno de:

40 (i) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la columna vecina izquierda;

definir un modo de intrapredicción representativo para la columna vecina izquierda como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la columna vecina izquierda; y

45 usar el modo de intrapredicción representativo definido para la columna vecina izquierda como el MPM para la columna vecina izquierda, o

50 (ii) contar, en unidades del tamaño de unidad de predicción más pequeño, modos de intrapredicción usados en una pluralidad de bloques vecinos en la fila vecina superior;

definir un modo de intrapredicción representativo para la fila vecina superior como el más frecuente de los modos de intrapredicción usados en la pluralidad de bloques vecinos en la fila vecina superior; y

55 usar el modo de intrapredicción representativo definido para la fila vecina superior como el MPM para la fila vecina superior;

codificar un elemento sintáctico que indica si se usa un índice MPM o un índice no MPM para indicar un modo de intrapredicción seleccionado de la pluralidad de modos de intrapredicción para la intrapredicción de un bloque actual de datos de vídeo;

60 codificar el índice indicado del índice MPM o el índice no MPM, donde el índice MPM indica cuál de los M MPM es el modo de intrapredicción seleccionado, y donde el índice no MPM indica cuál de la pluralidad de modos de intrapredicción distintos de los M MPM es el modo de intrapredicción seleccionado; y

65 codificar el bloque actual de datos de vídeo en función del modo de intrapredicción seleccionado.

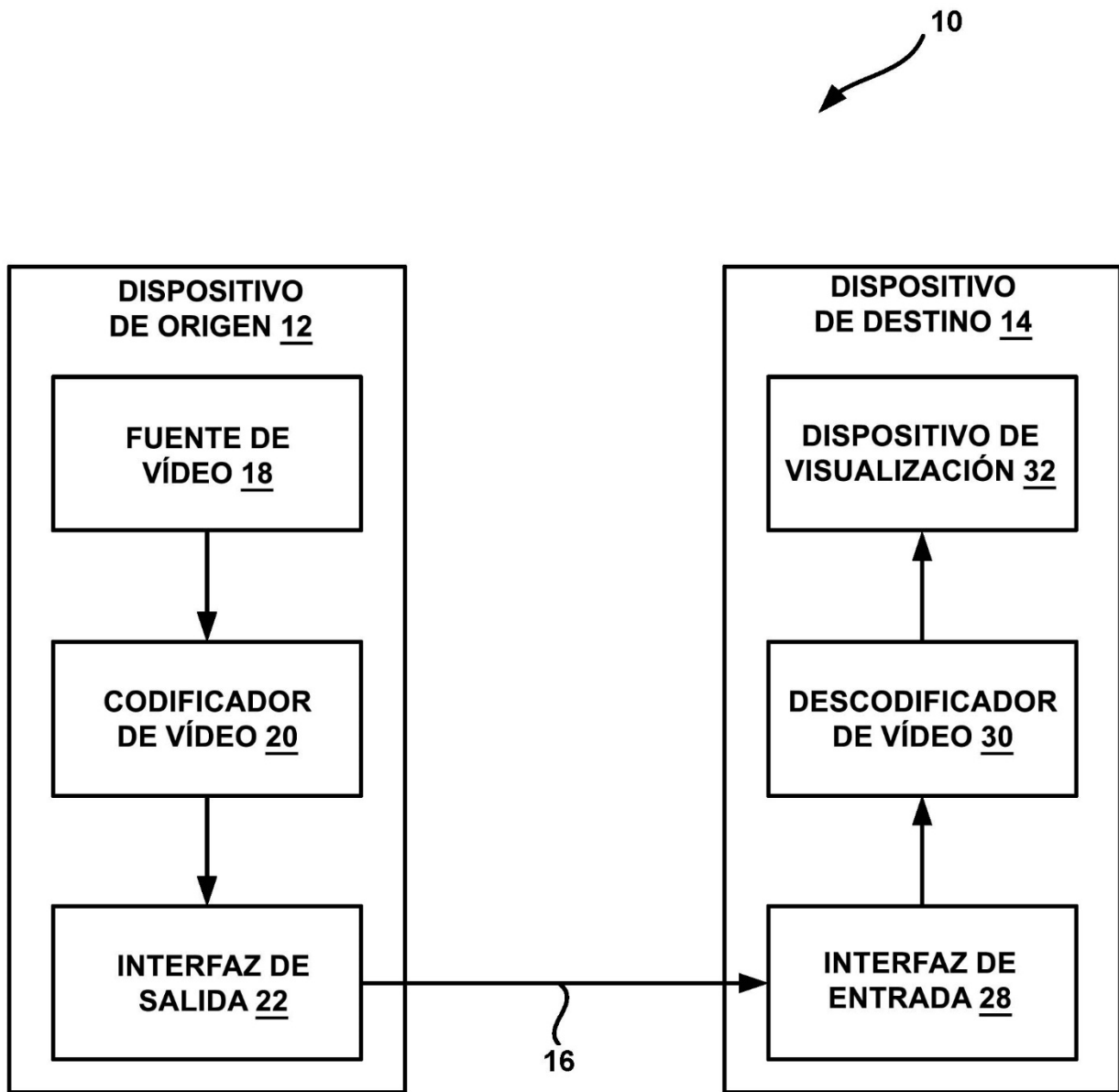


FIG. 1

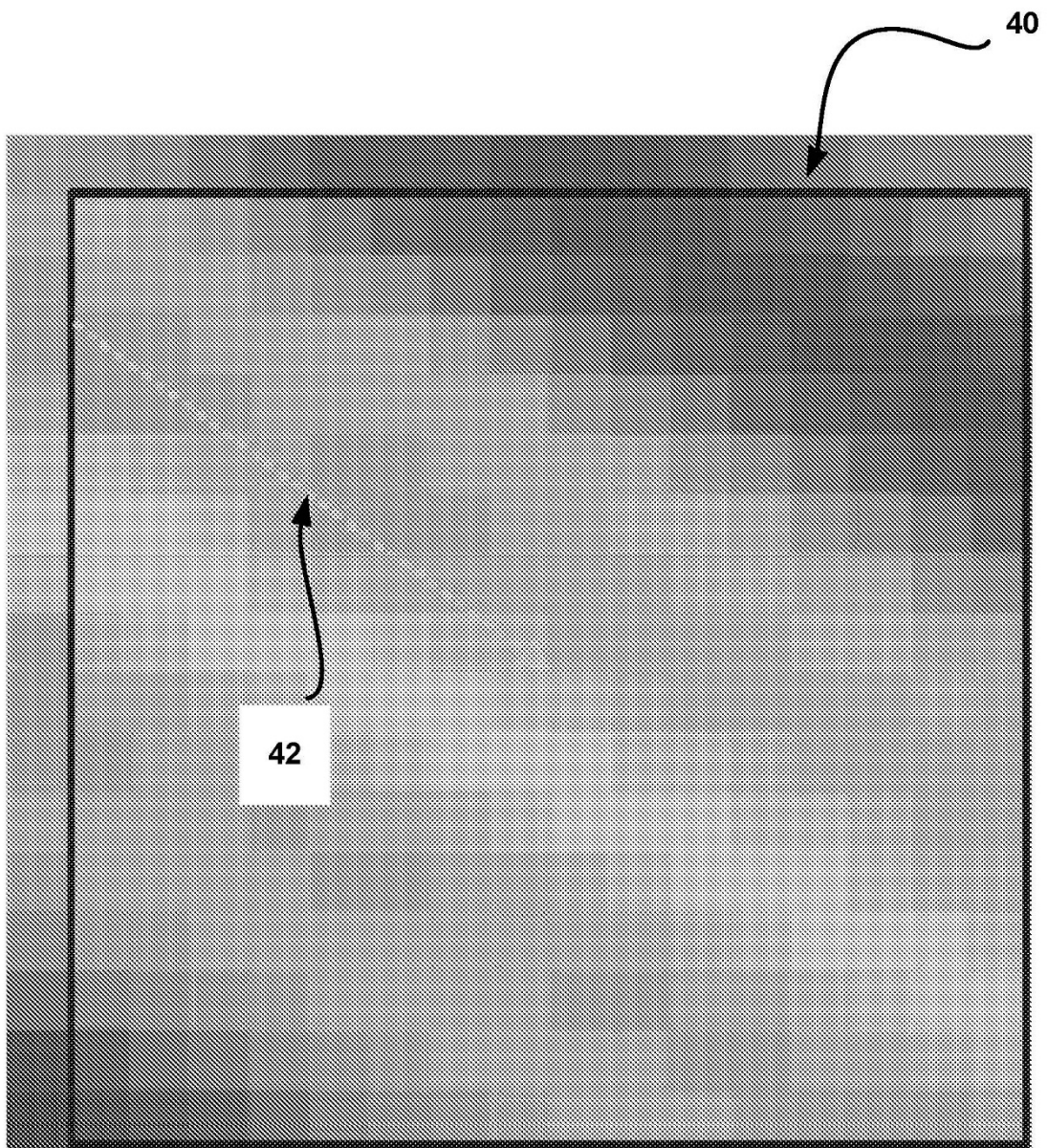


FIG. 2

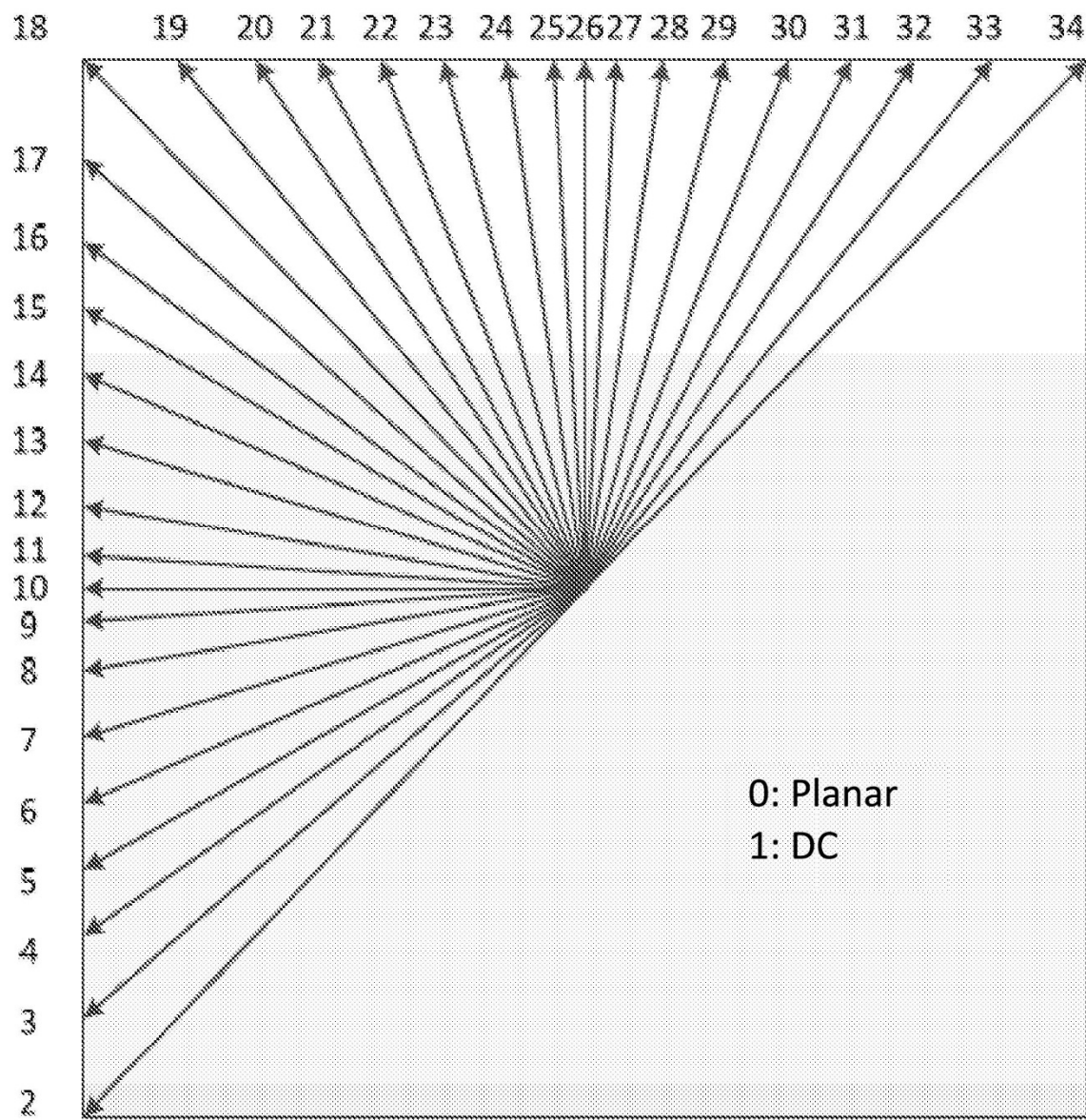


FIG. 3

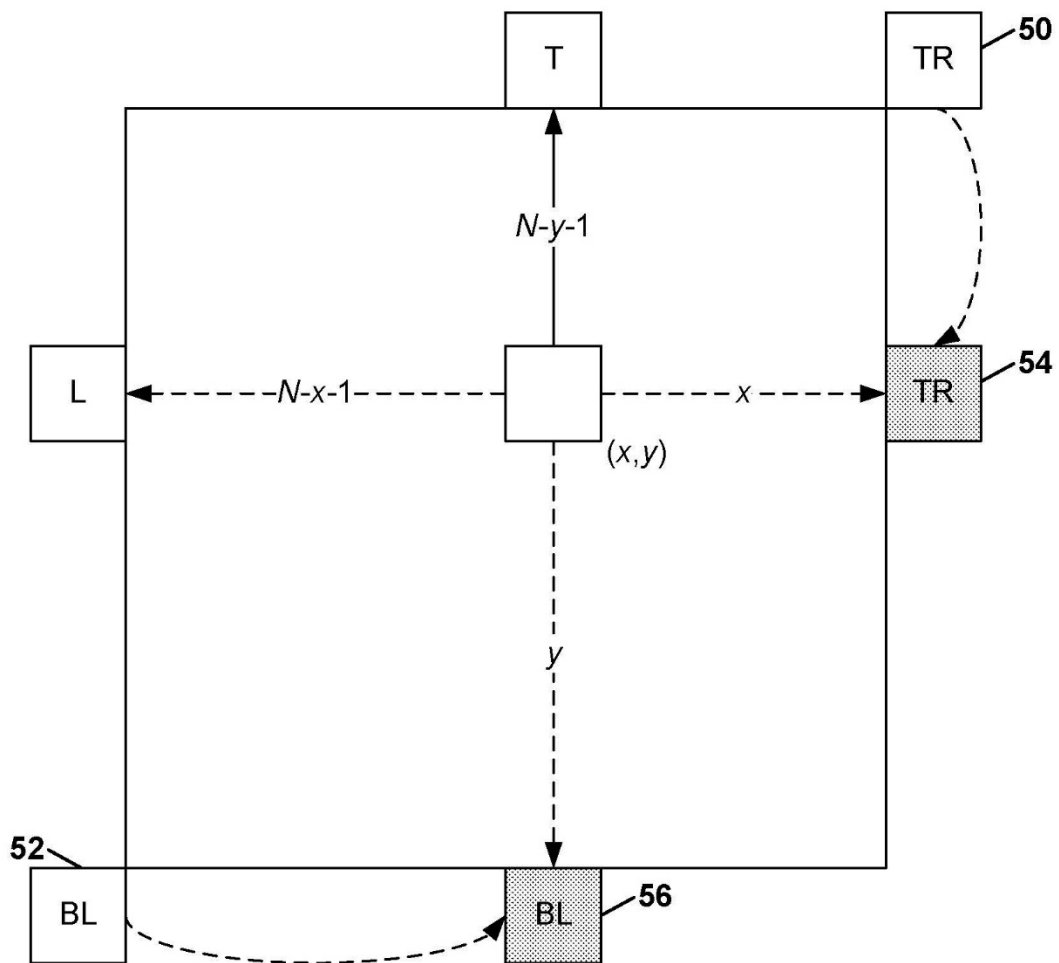


FIG. 4

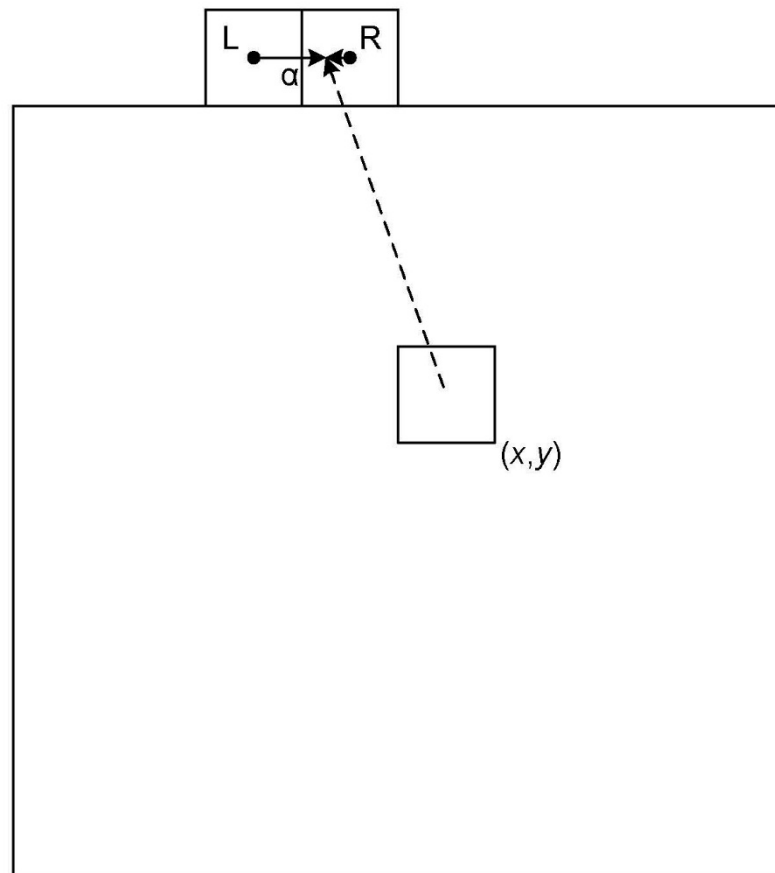


FIG. 5

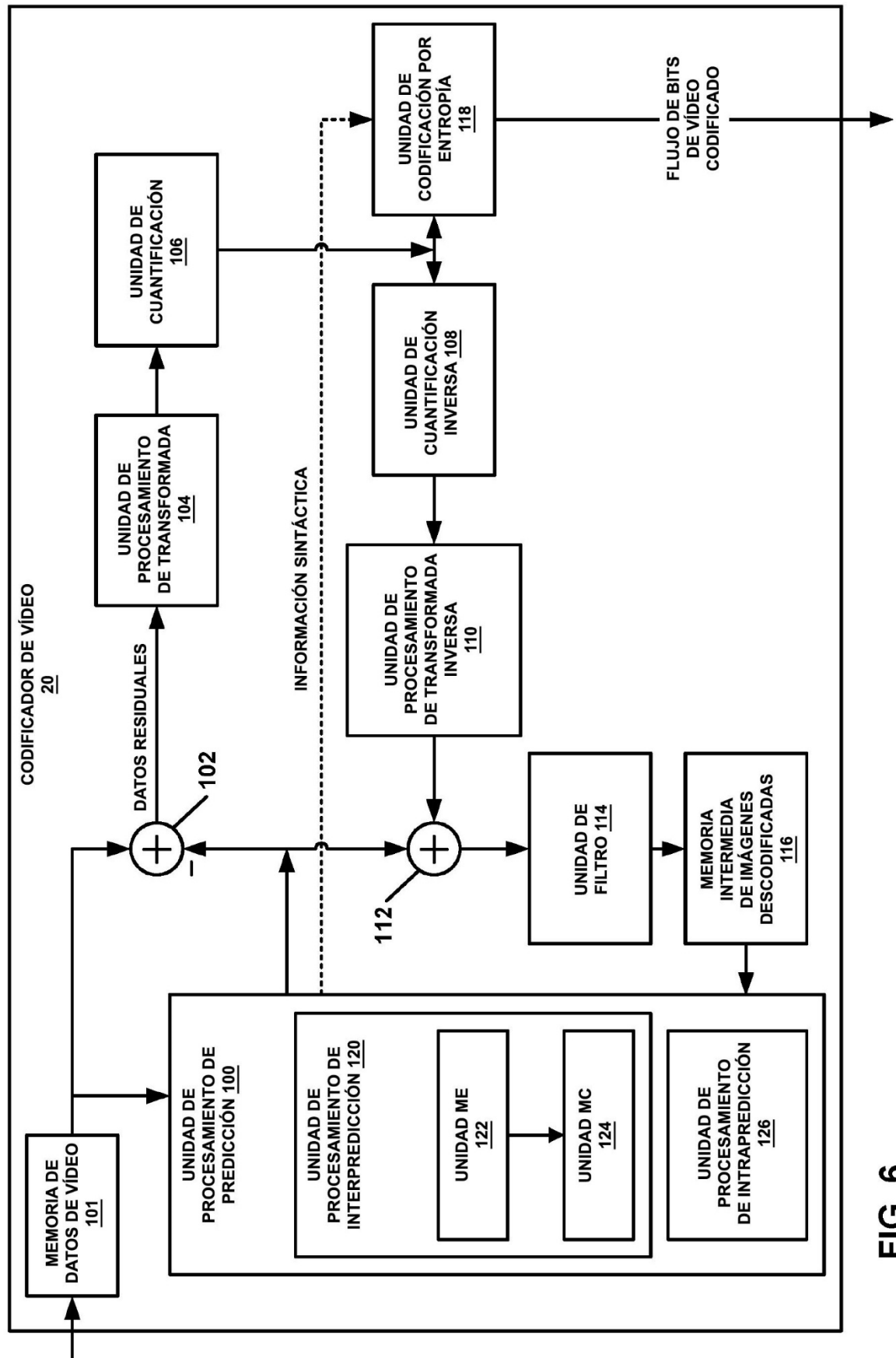


FIG. 6

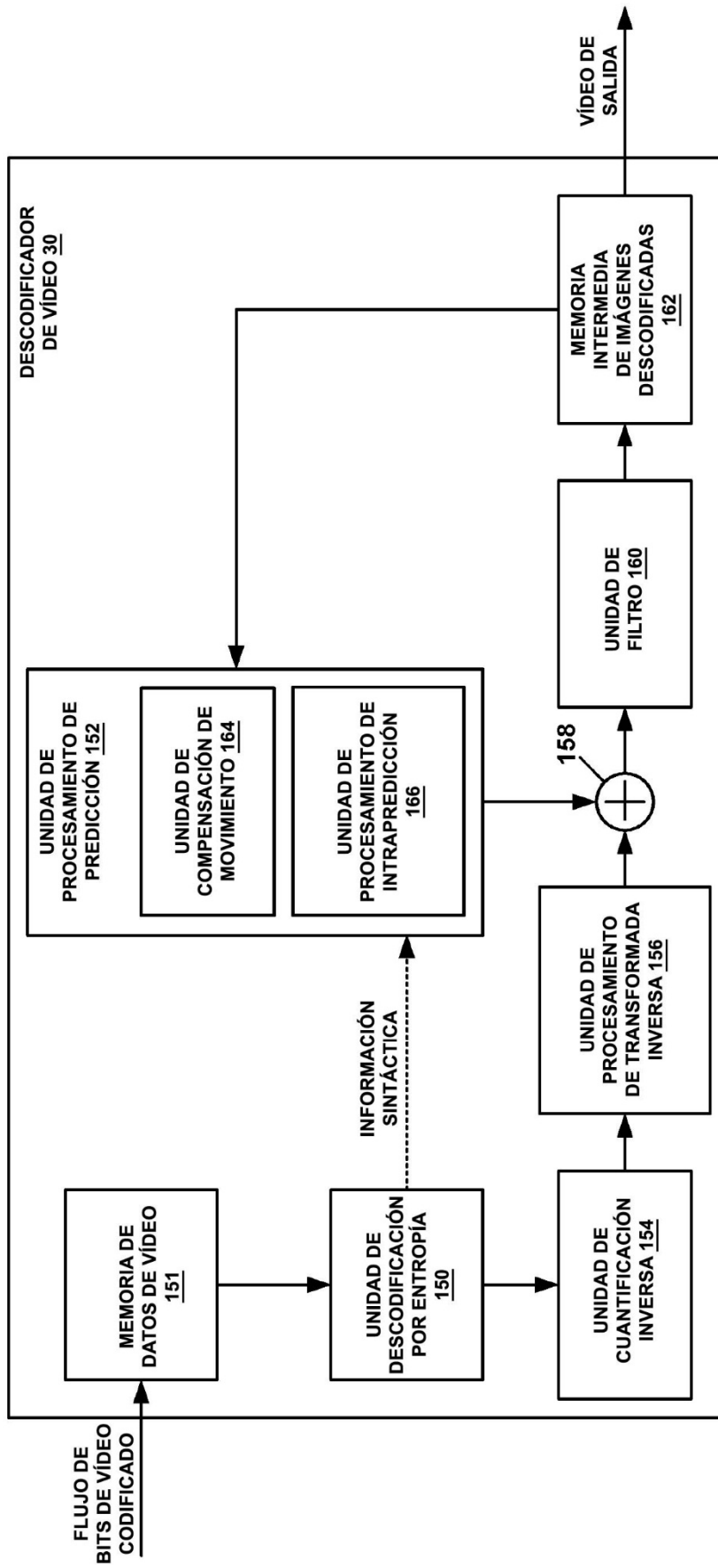


FIG. 7

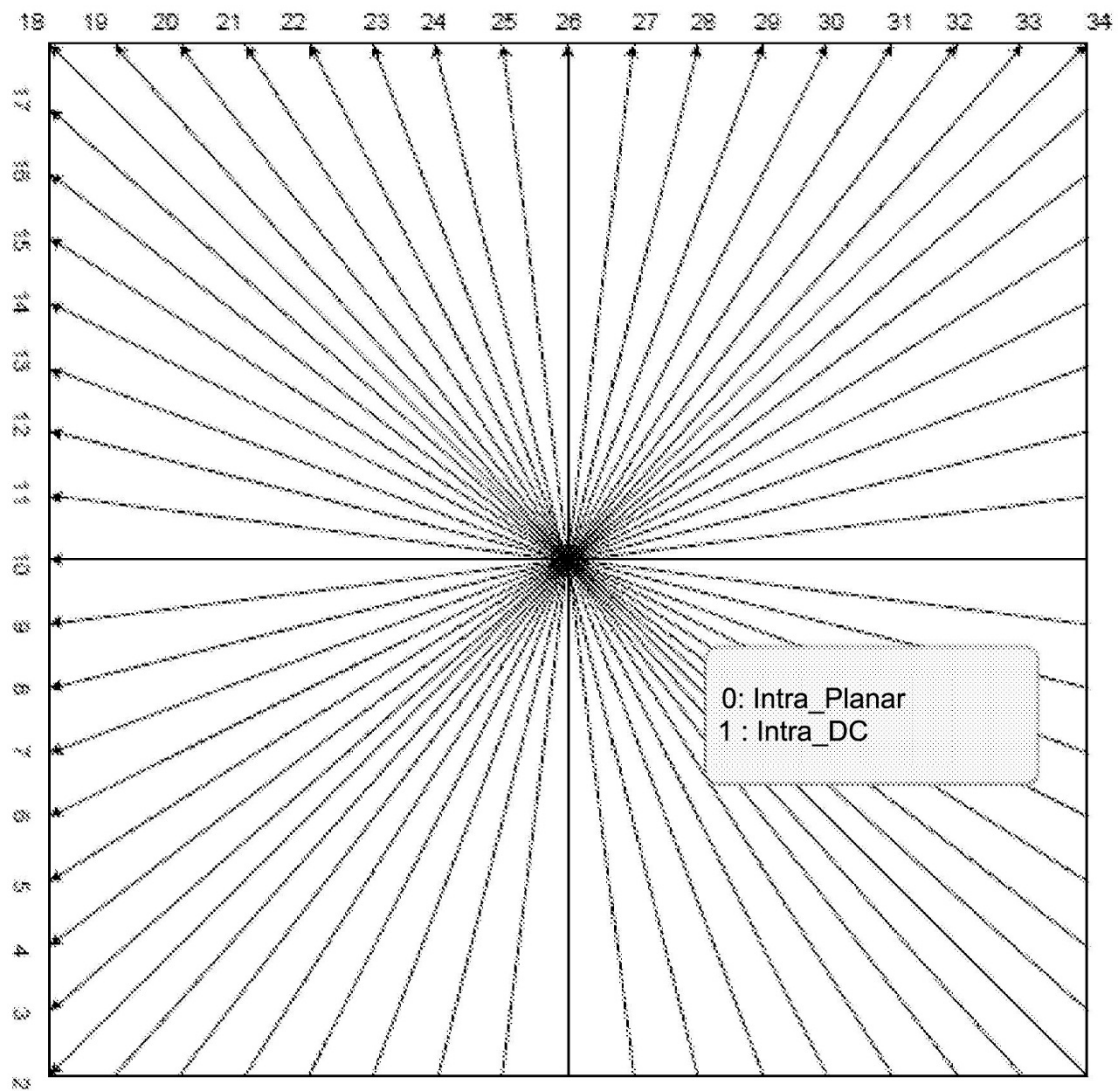


FIG. 8

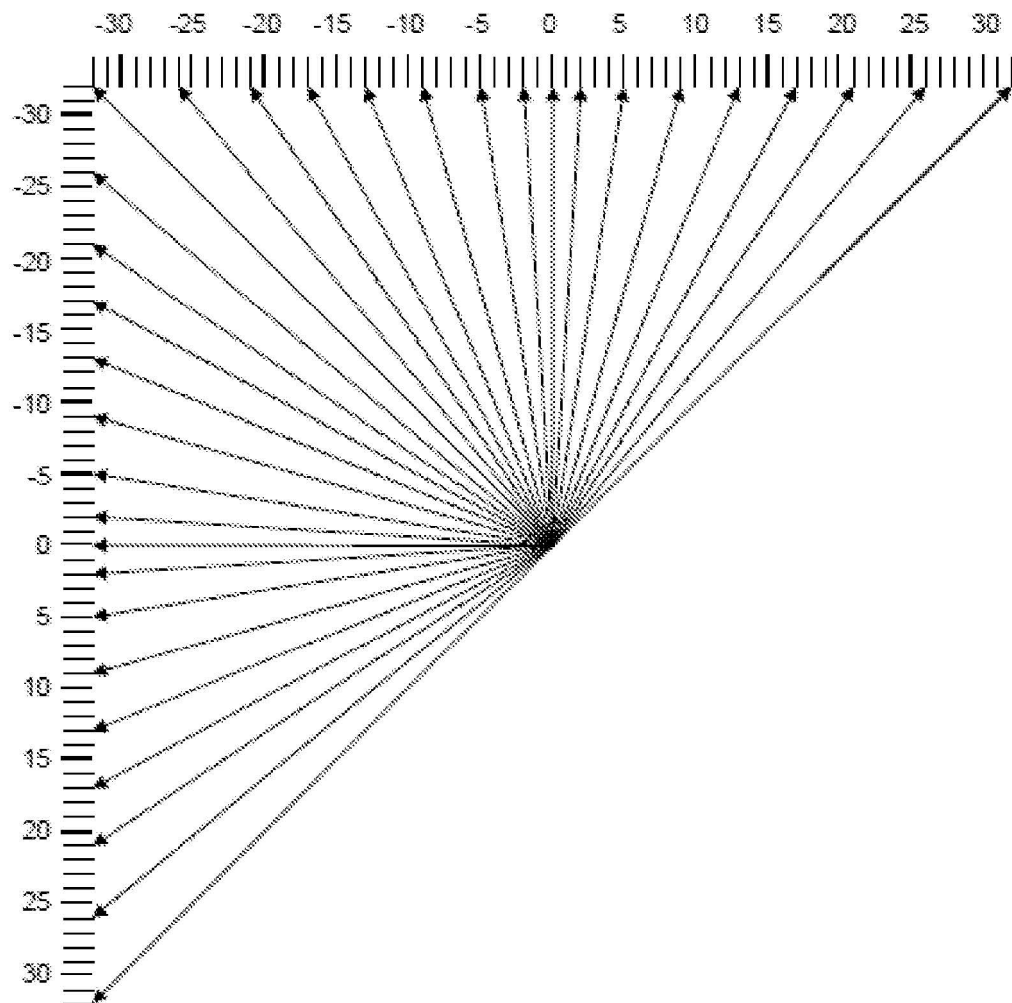


FIG. 9

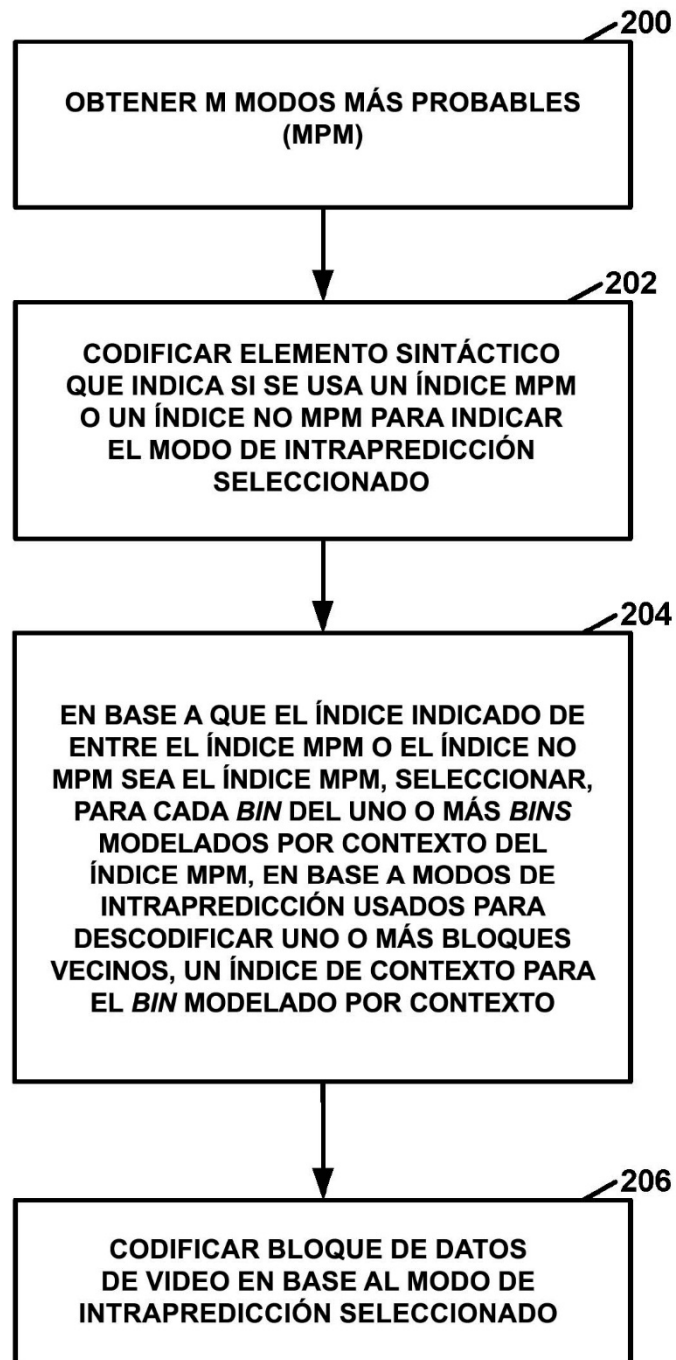


FIG. 10

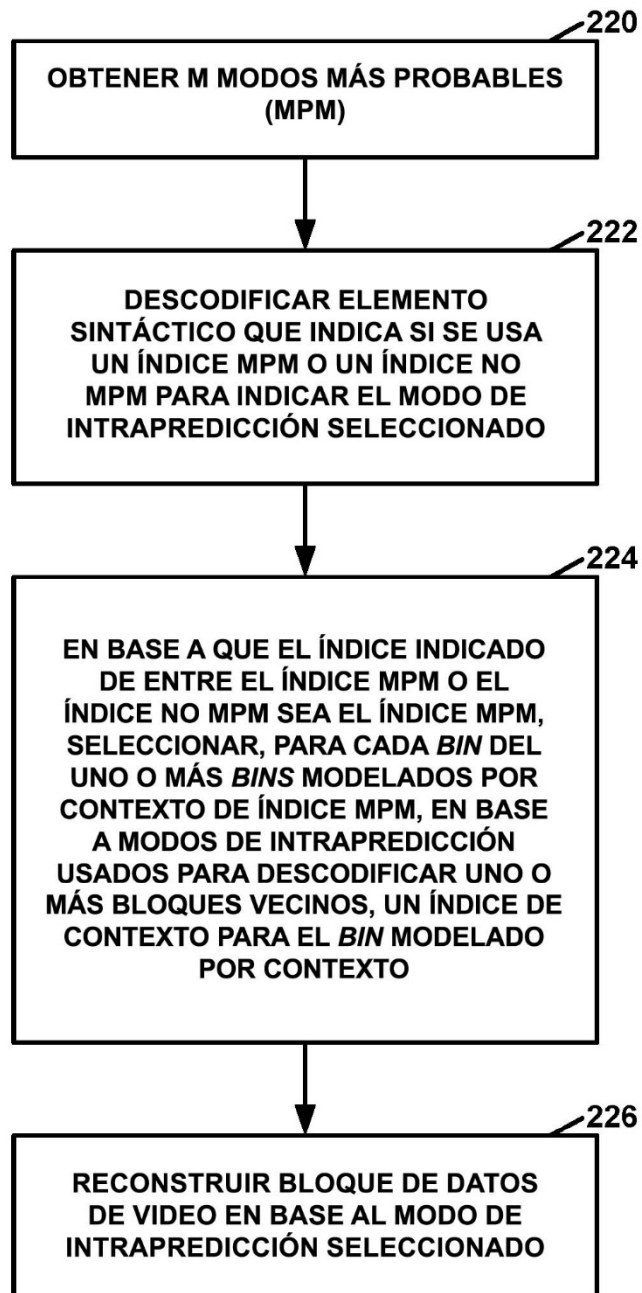


FIG. 11

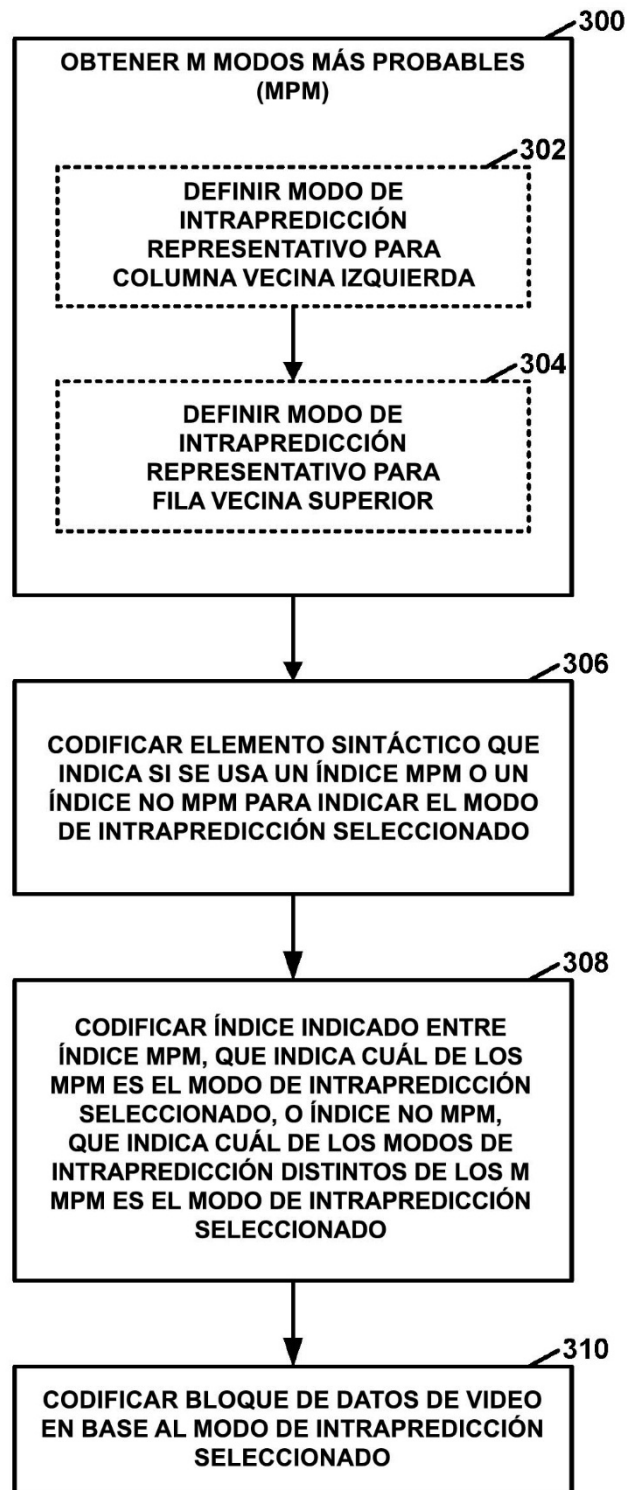


FIG. 12

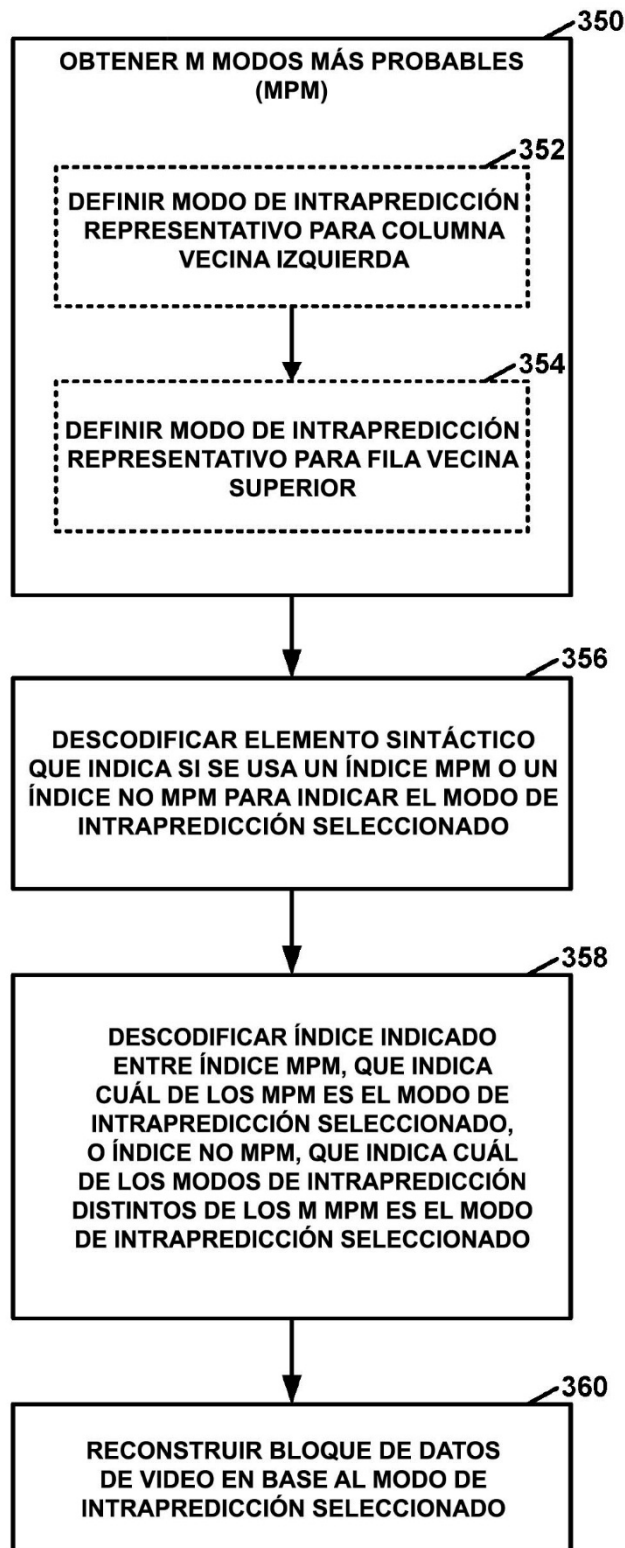


FIG. 13

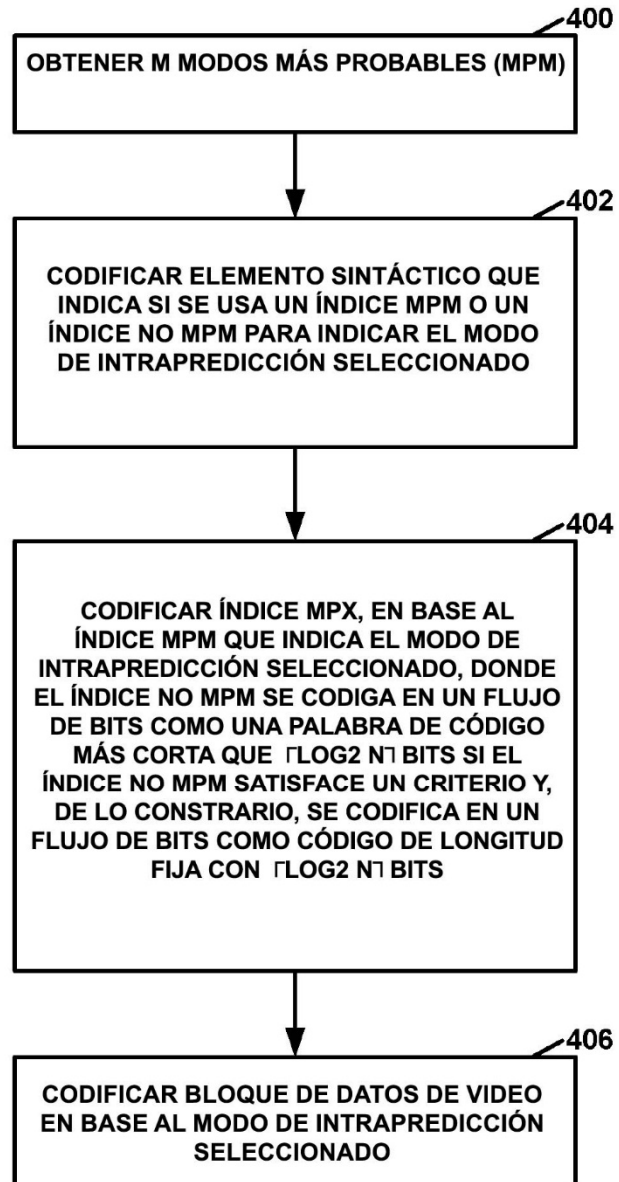


FIG. 14

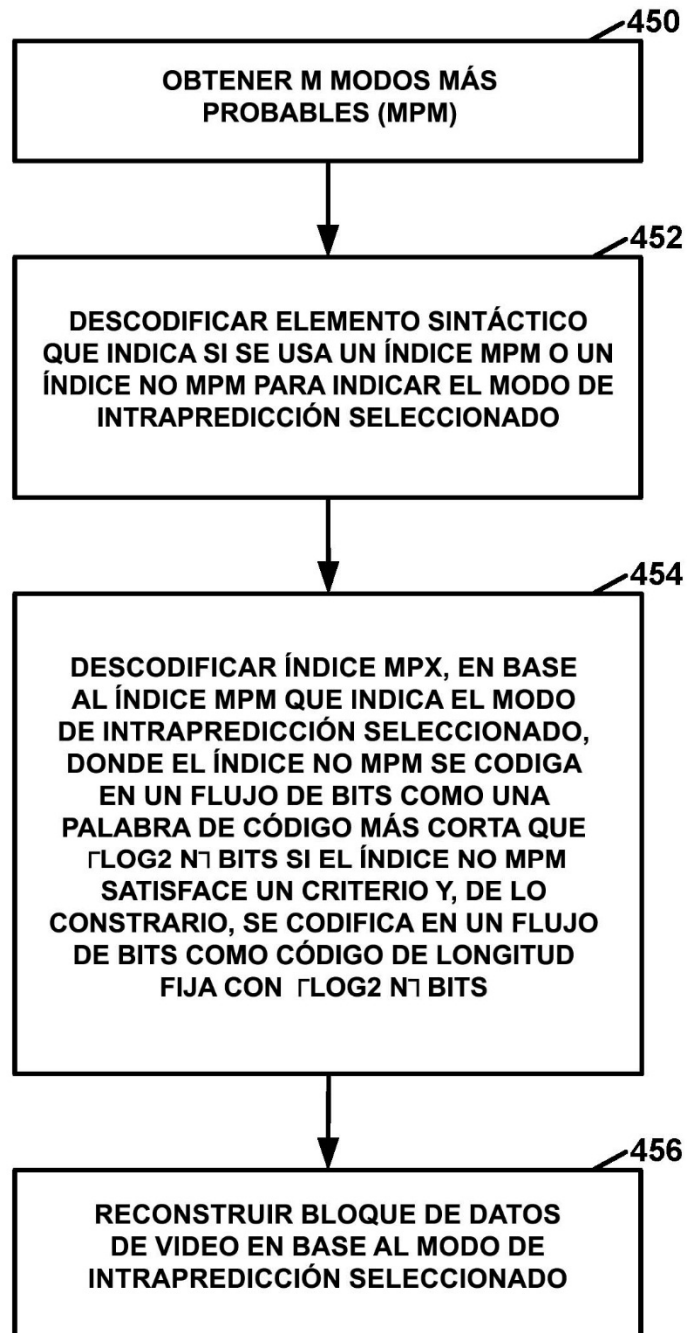


FIG. 15

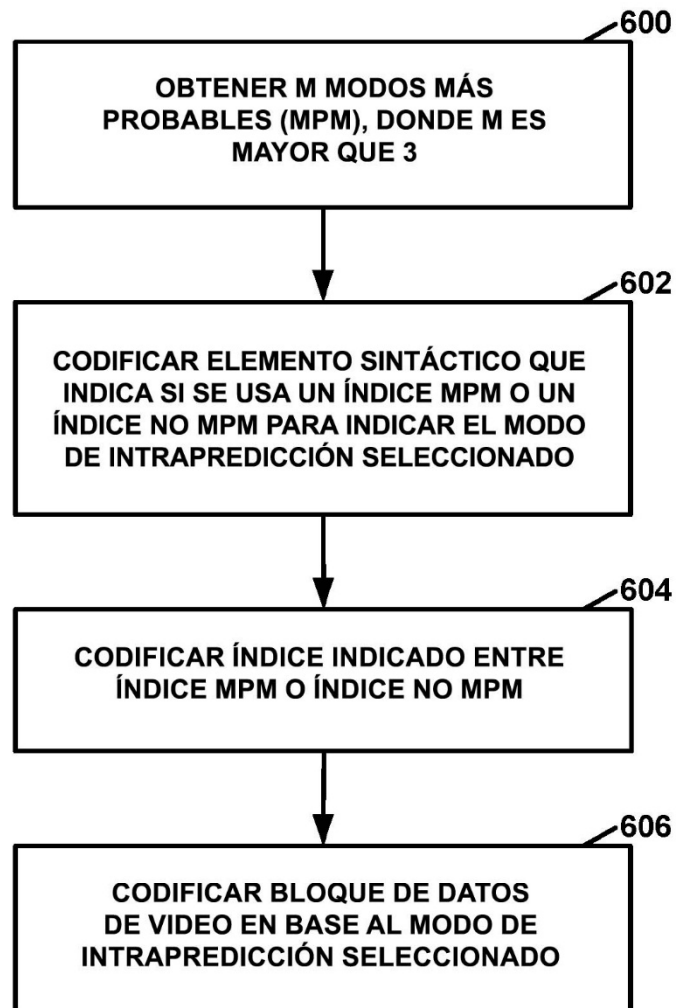


FIG. 16

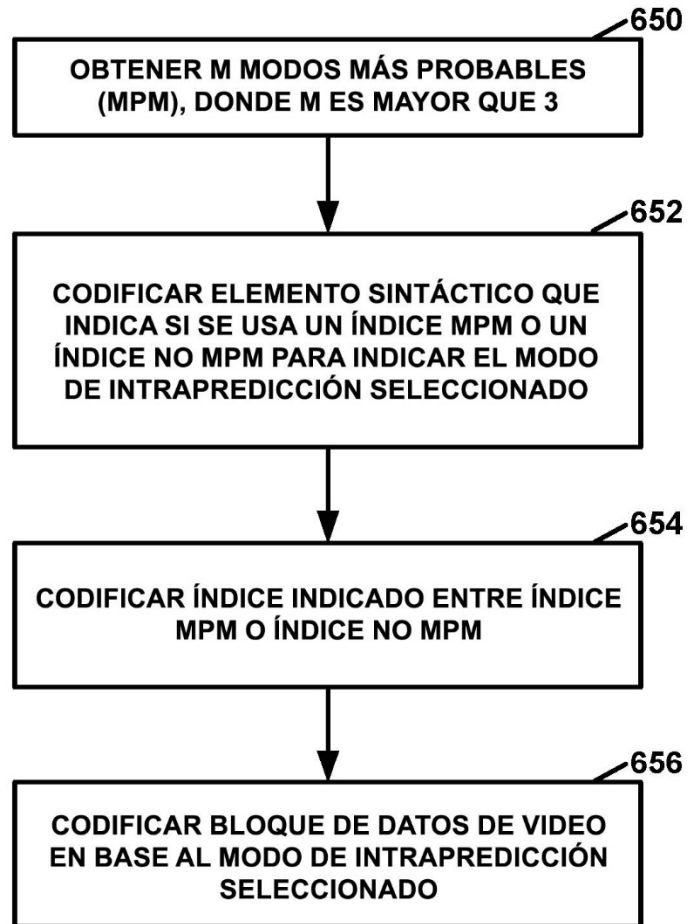


FIG. 17

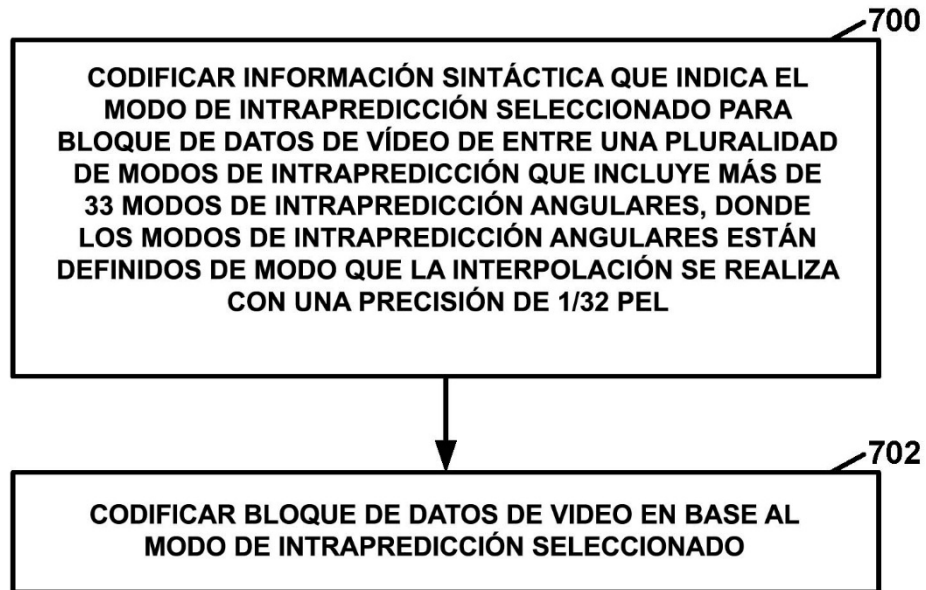


FIG. 18

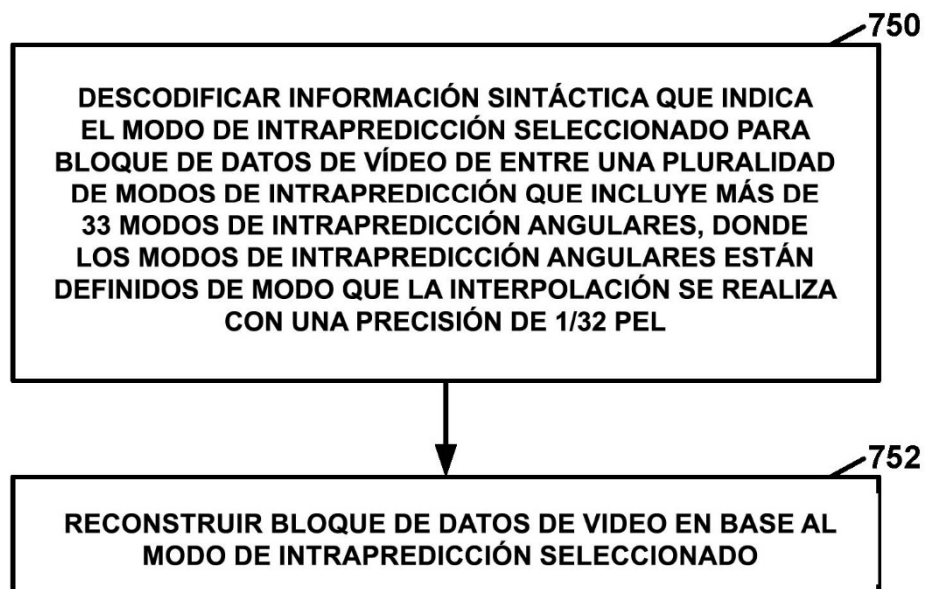


FIG. 19

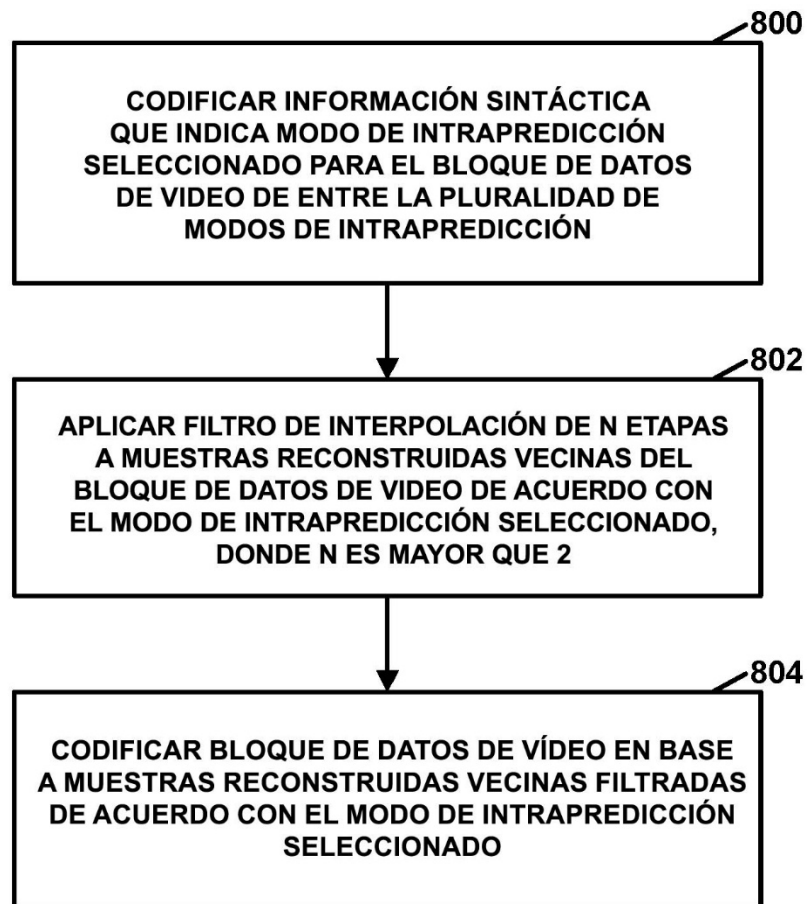


FIG. 20

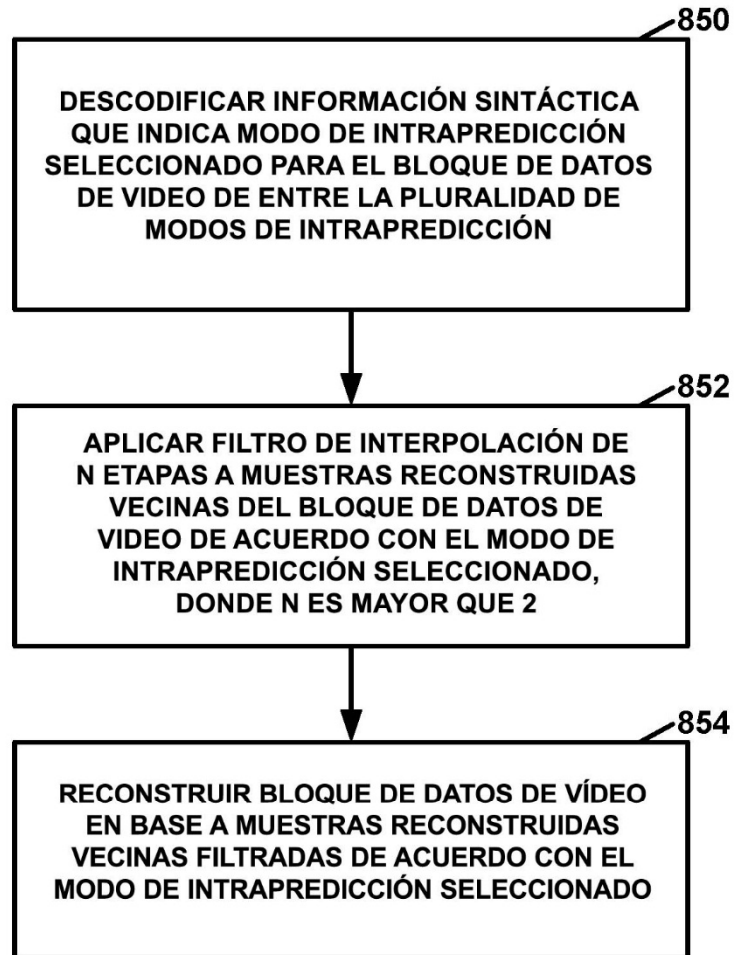


FIG. 21