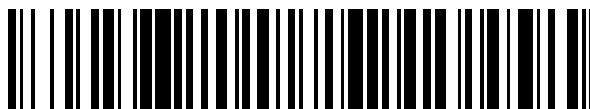


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 526 053**

51 Int. Cl.:

H04N 19/196	(2014.01) H04N 19/96	(2014.01)
H04N 19/13	(2014.01) H04N 19/194	(2014.01)
H04N 19/132	(2014.01) H04N 19/11	(2014.01)
H04N 19/14	(2014.01) H04N 19/593	(2014.01)
H04N 19/129	(2014.01) H04N 19/46	(2014.01)
H04N 19/122	(2014.01) H04N 19/157	(2014.01)
H04N 19/176	(2014.01) H04N 19/85	(2014.01)
H04N 19/159	(2014.01)	
H04N 19/60	(2014.01)	
H04N 19/61	(2014.01)	

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **08.07.2011** **E 11733763 (4)**

97 Fecha y número de publicación de la concesión europea: **12.11.2014** **EP 2591600**

54 Título: **Adaptación del conjunto de posibles transformaciones de frecuencia en base al tamaño del bloque y al modo intra**

30 Prioridad:

07.07.2011 US 201113178438
07.07.2011 US 201113178427
07.07.2011 US 201113178434
26.08.2010 US 201061377420 P
09.07.2010 US 201061363127 P
17.08.2010 US 201061374573 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
05.01.2015

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121, US

72 Inventor/es:

KARCZEWICZ, MARTA;
COBAN, MUHAMMED ZEYD;
CHEN, PEISONG;
CHUANG, HSIAO-CHIANG y
JOSHI, RAJAN LAXMAN

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 526 053 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

Adaptación del conjunto de posibles transformaciones de frecuencia en base al tamaño del bloque y al modo intra

Campo técnico

La presente divulgación se refiere a la codificación de vídeo.

5 Antecedentes

Las capacidades de vídeo digital se pueden incorporar en un amplio rango de dispositivos, incluyendo televisores digitales, sistemas digitales de transmisión directa, sistemas de transmisión inalámbricos, asistentes digitales personales (PDA), ordenadores portátiles o de sobremesa, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos de radio celulares o vía satélite, dispositivos de teleconferencia de vídeo, y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como las descritas en los estándares definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación avanzada de vídeo (AVC), la próximo estándar de Codificación de vídeo de alta eficiencia (HEVC) (también conocido como H.265), y extensiones de dichos estándares, para transmitir y recibir información de vídeo digital de una manera más eficiente.

Las técnicas de compresión de vídeo realizan predicción espacial y/o predicción temporal para reducir o eliminar la redundancia inherente en las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fotograma o sección de vídeo puede dividirse en macrobloques. Cada macrobloque se puede dividir aún más. Los macrobloques en un (I) fotograma o sección intracodificados se codifican utilizando predicción espacial respecto a los macrobloques vecinos. Los macrobloques en un (P o B) fotograma o porción intracodificados pueden usar predicción espacial respecto a los macrobloques vecinos en el mismo fotograma o porción o predicción temporal respecto a otros fotogramas de referencia.

Se hace referencia a los siguientes documentos de la técnica anterior:

DAVIES (BBC) T et al: "Suggestion for a Test Model", 1. REUNIÓN JCT-VC; 15-4-2010 - 23-4-2010; DRESDEN; (EQUIPO COLABORATIVO CONJUNTO EN CODIFICACIÓN DE VIDEO DE ISO/IEC JTC 1/SC29/WG11 E ITU-TSG.16); URL: [HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/](http://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/), 7 de mayo de 2010 (2010-05-07) divulga una sugerencia para un modelo de prueba para el próximo HEVC, en la que se proporciona un número diferente de transformaciones dependiendo del tamaño de un bloque.

XIN ZHAO ET AL: "Rate-distorsion optimized transform for intra-frame coding", ACOUSTICS SPEECH AND SIGNAL PROCESSING (ICASSP), CONFERENCIA INTERNACIONAL IEEE 2010, IEEE, Piscataway, NJ, EE.UU., 14 de marzo de 2010 (2010-03-14), páginas 1414-1417, ISBN: 978-1-4244-4295-9, divulga la asociación de varias transformaciones a cada modo de predicción interna, seleccionando una de las mismas utilizando un criterio de tipo de distorsión e indicando la transformación seleccionada al decodificador.

Sumario

La presente invención se refiere a un procedimiento de decodificación de datos de vídeo de acuerdo con la reivindicación 1 adjunta, a un aparato para decodificar datos de vídeo de acuerdo con la reivindicación 7 adjunta, a un producto de programa de ordenador de acuerdo con la reivindicación 9 adjunta, a un procedimiento de codificación de datos de vídeo de acuerdo con la reivindicación 10 adjunta, a un aparato para codificar datos de vídeo de acuerdo con la reivindicación adjunta 13 y a un producto de programa de ordenador de acuerdo con la reivindicación 15 adjunta.

40 Breve descripción de los dibujos

La figura 1 es un diagrama de bloques que ilustra un ejemplo de un sistema codificación y decodificación de vídeo que puede utilizar técnicas para la codificación y decodificación de unidades transformaciones de una unidad de codificación.

La figura 2 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar cualquiera o todas las técnicas para la codificación de unidades transformaciones de datos de vídeo descritas en esta divulgación.

La figura 3 es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo, que decodifica una secuencia de vídeo codificada.

La figura 4 es un diagrama conceptual que ilustra un gráfico que representa un conjunto de ejemplo de direcciones de predicción asociadas con varios modos de predicción interna.

La figura 5 es un diagrama conceptual que ilustra varios modos de predicción interna especificados por el estándar ITU-T H.264/AVC.

La figura 6 es un diagrama conceptual que ilustra una exploración en zigzag para un bloque de coeficientes de transformaciones.

Las figuras 7A y 7B son diagramas conceptuales que ilustran un árbol cuádruple de ejemplo y una unidad de

codificación más grande (LCU) correspondiente.

La figura 8 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la selección de una transformación y una exploración para aplicarse a un bloque basado en un modo de predicción interna seleccionado para el bloque.

La figura 9 es un diagrama de flujo que ilustra otro ejemplo de procedimiento para la selección de una transformación y una exploración para aplicarse a un bloque basado en un modo de predicción interna seleccionado para el bloque.

La figura 10 es un diagrama de flujo, que ilustra un ejemplo de procedimiento para aplicar un modo de predicción interna y una transformación a sub-UCs de tamaño particular.

La figura 11 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la realización de una exploración adaptativa de coeficientes de transformación basados en una transformación seleccionada aplicada a un bloque de valores residuales.

La figura 12 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la selección de un modelo de contexto a utilizar en la exploración y elementos de sintaxis de codificación de entropía que describen coeficientes adaptativos escaneados.

La figura 13 es un diagrama de flujo que ilustra un ejemplo de procedimiento para decodificar una unidad de transformación que se ha codificado utilizando una o más de las técnicas de esta divulgación.

La figura 14 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la selección de una transformación para aplicar a un bloque intracodificado que incluye un borde para el que se indica el modo de predicción interna DC.

Descripción detallada

En general, la presente divulgación describe técnicas para la codificación de datos de vídeo. Más específicamente, esta divulgación describe técnicas relativas a la transformación de datos residuales y a la exploración de coeficientes de transformación durante un proceso de codificación de vídeo. Los datos de vídeo codificados pueden incluir datos de predicción y datos residuales. Un codificador de vídeo puede producir los datos de predicción durante un modo de intra-predicción o un modo de inter-predicción. La intra-predicción general consiste en predecir un bloque de una imagen en relación con bloques vecinos previamente codificados de la misma imagen. La inter-predicción generalmente implica la predicción de un bloque de una imagen relativa a los datos de una imagen previamente codificada.

Después de una intra- o inter-predicción, un codificador de vídeo puede calcular un valor residual para el bloque. El valor residual generalmente corresponde a la diferencia entre los datos previstos para el bloque y el verdadero valor del bloque. Para comprimir aún más el valor residual de un bloque, el valor residual puede transformarse en un conjunto de coeficientes de transformación que compacta tantos datos (también denominados como "energía") como sea posible en el menor número de coeficientes como sea posible. Los coeficientes de transformación corresponden a una matriz bidimensional de coeficientes que es del mismo tamaño que el bloque original. En otras palabras, hay justo tantos coeficientes de transformación como píxeles en el bloque original. Sin embargo, debido a la transformación, muchos de los coeficientes de transformación pueden tener valores iguales a cero.

En algunos casos, una transformación secundaria, tal como una transformación de rotación, se puede aplicar a un subconjunto de los coeficientes de transformación generados por la primera transformación. Por ejemplo, después de transformar un bloque residual de 16 x 16 en una matriz de 16 x 16 de coeficientes de transformación, una transformación de rotación puede aplicarse al bloque de 8 x 8 de coeficientes de transformación de frecuencia más baja. Aunque este ejemplo describe una transformación de rotación como una transformación secundaria, otras transformaciones secundarias (por ejemplo, KLTS, DCT, y similares) pueden también ser aplicadas como transformaciones secundarias. Tales transformaciones secundarias también pueden seleccionarse en base a un modo de intra-predicción señalado para el bloque.

Las referencias a "transformaciones DCT" deben entenderse que incluyen ambas implementaciones de punto fijo e implementaciones de punto flotante. Es decir, una implementación de una transformación DCT puede comprender realmente una aproximación de una DCT, de tal manera que la transformación DCT tiene coeficientes enteros (es decir, coeficientes de punto fijo) en lugar de coeficientes de números racionales.

En algunos ejemplos, una transformación puede comprender una transformación no separable. Las transformaciones no separables son típicamente computacionalmente caras, y por lo tanto, los dispositivos de codificación de vídeo pueden aplicar transformaciones separables en su lugar. En general, las transformaciones separables incluyen un componente horizontal aplicado a las filas del bloque y un componente vertical aplicado a las columnas del bloque. De esta manera, una transformación separable puede tener un componente de transformación de fila y un componente de transformación de columna, también llamados como dos componentes ortogonales de la transformación. Dos matrices pueden ser utilizadas para definir una transformación separable, cada una de las matrices correspondiente a uno de los componentes ortogonales de la transformación. Una transformación no separable puede incluir sólo una matriz que, cuando se aplica, produce un resultado conceptualmente similar a la aplicación de la transformación separable, pero a través de cálculos relativamente más intensivos.

La transformación de un bloque de datos residuales produce un conjunto de coeficientes de transformación para el bloque. El codificador de vídeo puede entonces cuantificar los coeficientes de transformación para comprimir los datos de vídeo. La cuantificación generalmente implica valores de mapeo dentro de un conjunto relativamente grande de valores en un conjunto relativamente pequeño, reduciendo así la cantidad de datos necesarios para representar los coeficientes cuantificados de la transformación. Tras la cuantificación, el codificador de vídeo puede explorar los coeficientes de transformación, producir un vector unidimensional de la matriz de dos dimensiones incluyendo los coeficientes de transformación cuantificados. El codificador de vídeo puede ajustar a cero determinados coeficientes antes o después de la exploración, por ejemplo, todos menos la esquina superior izquierda de la matriz o todos los coeficientes de la matriz desde una posición N al final de la matriz.

El codificador de vídeo puede entonces codificar la entropía de la matriz resultante, para comprimir aún más los datos. En algunos ejemplos, el codificador de vídeo puede estar configurado para utilizar códigos de longitud variable (VLC) para representar varios posibles coeficientes de transformación cuantificados de la matriz, por ejemplo, utilizando codificación de longitud variable adaptativa al contexto (CAVLC). En otros ejemplos, el codificador de vídeo puede configurarse para usar aritmética binaria de codificación para codificar los coeficientes cuantificados resultantes, por ejemplo, usando codificación aritmética binaria adaptativa al contexto (CABAC).

La presente divulgación describe varias técnicas relacionadas con transformación, cuantificación, exploración, y codificación de entropía de valores residuales durante un proceso de codificación de vídeo. Las técnicas pueden ser aplicadas a la codificación y a la decodificación de vídeo, incluyendo codificadores /decodificadores (CODECs) de vídeo y unidades de procesamiento configuradas para realiza la codificación y/o la decodificación de vídeo. Las referencias a "unidades de codificación de vídeo" o "dispositivos de codificación de vídeo" deben entenderse para referirse a las unidades o dispositivos capaces de codificación, decodificación, o codificación y decodificación de datos de vídeo.

Se están realizando actualmente esfuerzos para desarrollar un nuevo estándar de codificación de vídeo, actualmente denominado Codificación de video de alta eficiencia (HEVC). El próximo estándar también se conoce como H.265. Los esfuerzos de estandarización se basan en un modelo de un dispositivo de codificación de vídeo conocido como Modelo de prueba HEVC (HM). El HM presume de varias capacidades de los dispositivos de codificación de vídeo sobre los dispositivos de acuerdo con, por ejemplo, ITU-T H.264/AVC. Por ejemplo, mientras que H.264 ofrece nueve modos de codificación de intra-predicción, HM ofrece nada menos que treinta y cuatro modos de codificación de intra-predicción.

El HM se refiere a un bloque de datos de vídeo como una unidad de codificación (CU), que puede incluir una o más unidades de predicción (PUs) y/o una o más unidades de transformación (TUs). Los datos de sintaxis dentro de un flujo de bits pueden definir una unidad de codificación más grande (LCU), que es una unidad de codificación más grande en términos de número de píxeles. En general, una CU tiene un propósito similar a un macrobloque de H.264, excepto que una CU no tiene una distinción de tamaño. Por lo tanto, una CU se puede dividir en sub-UCs. En general, las referencias en esta divulgación a una CU pueden referirse a una unidad de codificación más grande de una imagen o una sub-CU de una LCU. Una LCU puede dividirse en sub-UCs, y cada sub-CU puede dividirse aún más en sub-UCs. Los datos de sintaxis para un flujo de bits pueden definir un número máximo de veces que una LCU puede dividirse, a que se refiere como la profundidad de CU. En consecuencia, un flujo de bits también puede contaminar una unidad de codificación más pequeña (SCU). Esta divulgación también utiliza el término "bloque" para referirse a cualquiera de una CU, PU, o TU.

Una LCU puede estar asociada con una estructura de datos de árbol cuádruple. En general, una estructura de datos de árbol cuádruple incluye un nodo por CU, donde un nodo raíz corresponde a la LCU. Si una CU se divide en cuatro sub-CUs, el nodo correspondiente a la CU incluye cuatro nodos de hoja, cada uno de los cuales corresponde a una de las sub-UCs. Cada nodo de la estructura de datos de árbol cuádruple puede proporcionar datos de sintaxis para la CU correspondiente. Por ejemplo, un nodo de árbol cuádruple puede incluir una señal de división, que indica si la CU correspondiente al nodo se divide en sub-UCs. Los elementos de sintaxis de una CU pueden definirse de forma recursiva, y pueden depender de si la CU se divide en sub-CUs. Si una CU no se divide más, que se conoce como una CU de hoja. En esta divulgación, 4 sub-CUs de una CU de hoja también se conoce como CUs de hoja, aunque no hay división explícita de la CU de hoja original. Por ejemplo si una CU de tamaño 16 x 16 no se divide más, las cuatro sub-CUs de 8 x 8 también se conocen como CUs de hoja aunque la CU de 16 x 16 nunca se dividió.

Por otra parte, TUs de CUs de hoja también pueden estar asociadas con respectivas estructuras de datos de árbol cuádruple. Es decir, una CU de hoja puede incluir un árbol cuádruple que indica cómo se divide la CU de hoja en TUs. Esta divulgación se refiere al árbol cuádruple indicando cómo una LCU se reparte como árbol cuádruple CU y el árbol cuádruple indicando cómo se divide una CU de hoja en TUs como un árbol cuádruple TU. El nodo raíz de un árbol cuádruple TU generalmente corresponde a una CU de hoja, mientras que el nodo raíz de un árbol cuádruple CU corresponde generalmente a una LCU. Las TUs del árbol cuádruple TU que no se dividen se conocen como UTs de hoja.

Una CU de hoja puede incluir una o más unidades de predicción (PUs). En general, una PU representa toda o una porción de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Por ejemplo, cuando la PU está codificado en modo inter, la PU puede incluir datos que definen un vector de movimiento

de la PU. Los datos que definen el vector de movimiento puede describir, por ejemplo, un componente horizontal del vector de movimiento, un componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de píxel de un cuarto o precisión de píxel de un octavo), un fotograma de referencia al que apunta el vector de movimiento, y/o una lista de referencia (por ejemplo, lista 0 o lista 1) para el vector de movimiento. Los datos de la CU de hoja que definen la(s) PU(s) también pueden describir, por ejemplo, la partición de la CU en una o más PUs. Los modos de partición pueden ser diferentes en función de si la CU se codifica, se codifica en el modo de intra-predicción, o se codifica en el modo de inter-predicción. Para la codificación intra, una PU puede ser tratada de la misma como una unidad de transformación de hoja que se describe a continuación.

Una CU de hoja puede incluir una o más unidades de transformación (TUs). Las unidades de transformación pueden especificarse usando una estructura de árbol cuádruple TU, como se mencionó anteriormente. Es decir, una señal de división puede indicar si una CU de hoja se divide en cuatro unidades de transformación. Luego, cada unidad de transformación puede dividirse aún más en 4 subunidades de transformación. Cuando una TU no se divide más, puede llamarse como una TU de hoja. En general, para la codificación intra, toda las TUs de hoja pertenecientes a una parte de la CU de hoja comparten el mismo modo de predicción intra. Es decir, el mismo modo de intra-predicción se aplica generalmente para calcular los valores predichos para todas las TUs de una CU de hoja. Para codificación intra, un codificador de vídeo puede calcular un valor residual para cada TU de hoja usando el modo de predicción intra, como una diferencia entre la porción de los valores predictivos correspondientes a la TU y el bloque original. El valor residual puede ser transformado, cuantificado y escaneado. Para la codificación inter, un codificador de vídeo puede realizar la predicción a nivel de PU y puede calcular un residual para cada PU. Los valores residuales correspondientes a una CU de hoja se pueden transformar, cuantificar, y explorar. Para codificación inter, una hoja-de TU puede ser mayor o menor que una PU. Para la codificación intra, una PU puede colocarse con una correspondiente TU de hoja. En algunos ejemplos, el tamaño máximo de una TU de hoja puede ser el tamaño de la correspondiente CU de hoja.

En general, esta divulgación utiliza los términos CU y TU para referirse a la CU de hoja y la TU de hoja, respectivamente, a menos que se indique lo contrario. En general, las técnicas de esta divulgación se refieren a la transformación, cuantificación, exploración, y datos de codificación de entropía de una CU. Como ejemplo, las técnicas de esta divulgación incluyen la selección de una transformación para utilizarse para transformar un valor residual de un bloque de intra-predicción basado en un modo de intra-predicción utilizado para predecir el bloque. Esta divulgación también se utiliza el término "transformación direccional" o "transformación diseñada" para referirse a una transformación tal que depende de la dirección del modo de intra-predicción. Es decir, un codificador de vídeo puede seleccionar una transformación direccional a aplicar a una unidad de transformación (TU). Como se señaló anteriormente, la intra-predicción incluye la predicción de la TU de una CU actual de una imagen desde CUs y Tus previamente codificadas de la misma imagen. Más específicamente, un codificador de vídeo puede intra-predecir una TU actual de una imagen usando un modo de intra-predicción particular.

Las técnicas de la presente divulgación incluyen la asociación de ciertas transformaciones con los modos de intra-predicción. Por lo tanto, puede haber una correspondencia de uno a uno entre los modos de intra-predicción y las transformaciones de acuerdo con técnicas de esta divulgación. En algunos ejemplos, puede haber una correspondencia muchos a uno entre los modos de intra-predicción y sus transformaciones. Por ejemplo, un gran conjunto de modos de intra-predicción puede mapearse a un conjunto más pequeño de los modos de intra-predicción, y cada uno de los conjunto de modos de infra-predicción más pequeños se puede mapear uno a uno a las respectivas transformaciones.

Las transformaciones también se pueden mapear a patrones de exploración respectivos. En algunos ejemplos, modos de intra-predicción se pueden mapear a transformaciones y exploraciones, mientras que en otros ejemplos, modos de intra-predicción pueden correlacionarse con las transformaciones y las transformaciones se pueden mapear a las exploraciones. En varios ejemplos, se pueden utilizar varias combinaciones de transformaciones y exploraciones de coeficientes. Por ejemplo, los modos de intra-predicción pueden mapearse a transformaciones direccionales dependientes del modo, y una exploración en zigzag se puede utilizar en todos los casos.

En algunos ejemplos, en lugar de los modos de mapeo de intra-predicción para transformar y/o explorar patrones, el codificador de vídeo 20 puede estar configurado para indicar una combinación de una o más transformaciones y para aplicar un patrón de exploración. Asimismo, el decodificador de vídeo 30 puede estar configurado para determinar una transformación y una exploración del patrón aplicándose sobre la base de una indicación recibida, en lugar de un mapeo entre un modo de intra-predicción y las transformaciones y el patrón de exploración.

Las transformaciones pueden incluir una transformación discreta de coseno (DCT) y ocho transformaciones direccionales, también conocidas como transformaciones de Karhunen-Loeve (KLTs). La DCT es generalmente una suma de funciones coseno que tienen diferentes frecuencias, donde las funciones se aplican a los valores residuales. Las KLTs generalmente incluyen cada una dos matrices. Cada matriz en la KLT es del mismo tamaño que el bloque residual a transformar. Las KLTs pueden derivarse de datos del conjunto de entrenamiento o derivarse analíticamente asumiendo un modelo para los fotogramas de vídeo y/o predicción residual.

Un codificador HM puede estar configurado con treinta y cuatro modos de intra-predicción para ciertos tamaños de bloque. Por lo tanto, para apoyar un mapeo uno a uno entre los modos de intra-predicción direccionales y transformaciones direccionales, los codificadores y los decodificadores HM necesitarían almacenar hasta 68 matrices para cada tamaño de transformación soportado. Además, los tamaños de bloque para los que están soportados todos los treinta y cuatro modos de intra-predicción pueden ser bloques relativamente grandes, por ejemplo, 16 x 16 píxeles, 32 x 32 píxeles, o incluso mayores.

En algunos ejemplos, la presente divulgación proporciona técnicas para reducir el número de transformaciones direccionales que los codificadores y decodificadores deben soportar. Es decir, los codificadores y decodificadores pueden soportar menos transformaciones direccionales que el número de modos de intra-predicción disponibles. Un codificador de acuerdo con estas técnicas, por ejemplo, puede mapear un conjunto de modos de intra-predicción relativamente grande a un subconjunto de los modos de intra-predicción. Cada uno de los modos de intra-predicción en el subconjunto puede estar asociado con una transformación direccional. Es decir, los modos de intra-predicción en el subconjunto pueden tener una correspondencia de uno a uno con un conjunto de transformaciones direccionales. Por lo tanto, los modos de intra-predicción en el gran conjunto pueden tener una correspondencia de muchos a uno con el conjunto de transformaciones direccionales.

Por ejemplo, cada uno de los 34 modos de intra-predicción direccionales HM puede correlacionarse con uno de los ocho modos de intra-predicción direccionales de H.264. El codificador de vídeo, por lo tanto, puede seleccionar un modo de intra-predicción direccional para predecir un valor para una TU actual, determinar un modo de intra-predicción del subgrupo al que se mapea el modo seleccionado, y a continuación utilizar la transformación direccional mapeada al modo de intra-predicción del subconjunto para transformar la TU actual. Además, cada una de las transformaciones direccionales puede estar asociada con un respectivo patrón de exploración. Por lo tanto, el codificador puede realizar la exploración asociada con la transformación direccional para producir un vector de coeficientes de transformaciones que luego puede ser cuantificado. Por otra parte, el codificador puede estar configurado con un tamaño máximo para el vector. Es decir, el codificador puede detener la exploración de los coeficientes de transformación al alcanzar el tamaño máximo, si el siguiente coeficiente que se va a explorar es distinto de cero o no.

Mediante la aplicación de las técnicas descritas anteriormente, el codificador no necesita señalar la transformación utilizada para una TU particular, cuando se utilizan las técnicas descritas anteriormente. Es decir, el codificador y el decodificador pueden estar cada uno configurado con el mapeo de muchos a uno de modos de intra-predicción del gran conjunto de modos de intra-predicción del subconjunto, y con el mapeo uno a uno de modos de intra-predicción del subconjunto a transformaciones direccionales. Por lo tanto, mediante la señalización del modo de intra-predicción a partir del gran conjunto, el decodificador puede derivar la transformación utilizada para transformar el bloque. Por otra parte, estas técnicas pueden ser implementadas por los dispositivos legados que tienen una memoria limitada que se puede mapear para el almacenamiento de matrices para las diversas transformaciones direccionales.

Un codificador HM puede configurarse de manera que el conjunto disponible de modos de intra-predicción para un bloque varía en función del tamaño del bloque. Es decir, el tamaño de una CU puede determinar el número de modos de intra-predicción disponibles para la CU, de la que el codificador puede seleccionar un modo de intra-predicción para predecir los valores utilizados para calcular los coeficientes de las TUs. La tabla 1 a continuación ilustra un ejemplo de una correspondencia entre los tamaños de CU y el número de modos de intra-predicción disponibles para CUs de ese tamaño. En esta divulgación, 4 sub-CUs de una CU de hoja también se les conoce como CUs de hoja, aunque no hay una separación explícita de la CU de hoja original. Si la CU de hoja tiene el tamaño de CU más pequeño, estas 4 sub-CUs pueden seleccionar diferentes modos de predicción intra. Por lo tanto, la tabla tiene una entrada para el tamaño de CU de 4 x 4.

TABLA 1

Tamaño CU	Número de modos de intra-predicción
4 x 4	17
8 x 8	34
16 x 16	34
32 x 32	34
64 x 64	5

En general, un codificador de vídeo puede indicar una dirección de predicción para un bloque, para que un decodificador de vídeo decodifique correctamente el bloque. En algunos ejemplos, un codificador de vídeo puede estar configurado para determinar una dirección de predicción única para una CU que se puede aplicar a todas las TUs pertenecientes a la CU. Sin embargo, como se señaló anteriormente en la Tabla 1, ciertos tamaños de bloques tienen menos modos de intra-predicción disponibles en comparación con otros tamaños de bloques. Tales casos pueden resolverse permitiendo que el número de direcciones de predicción en el tamaño de bloque de CU que se utilizará para los tamaños de bloques de TU. Alternativamente, los modos de intra-predicción de un conjunto más

grande pueden ser mapeados a los modos de intra-predicción de un conjunto más pequeño, por ejemplo, un subconjunto. Como se describió anteriormente, puede haber una relación de muchos a uno entre los modos de intra-predicción del conjunto más grande y los modos de intra-predicción de un conjunto más pequeño.

Para estructuras de árbol cuádruple pueden conducir a la descomposición de un gran bloque (CU) en bloques más pequeños (TUs). El modo de predicción espacial del bloque de raíz (para la CU) puede señalarse explícitamente en el flujo de bits. Los bloques resultantes de TU de árbol cuádruple (TUs) más pequeños pueden heredar sus modos de predicción del bloque de raíz del árbol cuádruple de TU (que corresponde a la CU). Sin embargo, el número de direcciones de predicción espacial soportadas por los bloques más pequeños (TUs) puede ser diferente del bloque de raíz (CU). Esto se puede resolver permitiendo más direcciones de predicción para los bloques más pequeños (TUs). Alternativamente, los modos de predicción de los bloques más pequeños (TUs) pueden derivarse del bloque raíz (CU) mediante una relación de mapeo de muchos a uno o de uno a uno de acuerdo con un criterio predeterminado, como minimizar la diferencia del ángulo de dirección de predicción entre la dirección de predicción intra para la CU y las direcciones de predicción soportadas en el bloque más pequeño. Las transformaciones direccionales y los patrones de exploración pueden seleccionarse en base a este mapeo.

De esta manera, el codificador de vídeo puede indicar una dirección de intra-predicción una vez para una CU. Suponiendo que la CU incluye una TU de un tamaño que no admite la dirección de intra-predicción señalada, el codificador de vídeo puede determinar el modo de intra-predicción para la TU en base al mapeo. Es decir, el codificador de vídeo puede intra-prededir un bloque predicho utilizado para calcular una TU utilizando el modo de intra-predicción del conjunto más pequeño al que se mapea el modo de intra-predicción señalado del conjunto más grande. Del mismo modo, un decodificador de vídeo puede incluir la misma configuración, de tal manera que el decodificador de vídeo puede determinar modos de intra-predicción para cada TU de una CU recibida. Alternativamente, el número de modos de predicción para una TU se puede aumentar para que coincida con el número de modos de predicción para la CU correspondiente.

En algunos ejemplos, para algunos modos de intra-predicción, varios archivos de transformación pueden ser posibles para TUs de tamaños particulares. En tales casos, un decodificador de vídeo podría no ser capaz de derivar la transformación a aplicar a la TU únicamente desde el modo de intra-predicción. Por lo tanto, el codificador de vídeo puede necesitar señalar la transformación a usar para unidades de formación de tamaños para los que varias transformaciones son posibles. En lugar de señalización de la transformación para cada una de tales TUs, esta información puede señalizarse a nivel de CU. En tal caso, esta transformación puede aplicarse a todas las TUs contenidas en la CU. Para TUs de tamaños para los que sólo una transformación se correlaciona con el modo de intra-predicción señalado, puede utilizarse el mapeo de transformación.

Por otra parte, la sintaxis que especifica la transformación sólo necesita estar presente si la CU incluye una TU de un tamaño para el que son posibles varias transformaciones. Para TUs para las que sólo una transformación posible, el codificador y decodificador de vídeo pueden determinar la transformación de uso en base al modo de intra-predicción seleccionado. Sin embargo, para TUs de un tamaño para el que varias transformaciones son posibles, el codificador de vídeo puede indicar explícitamente la transformación a utilizar para todas las TUs de tamaño similar en la CU, por ejemplo, mediante la señalización de la transformación a utilizar en la raíz del árbol cuádruple de la TU para la CU.

De esta manera, si un decodificador de vídeo se encuentra con una TU de un tamaño para el que varias transformaciones son posibles en base al modo de intra-predicción para la CU correspondiente a la TU, el decodificador puede determinar la transformación a aplicar en base a la señalización explícita. Para otra TU, el decodificador de vídeo puede utilizar la transformación asociada con el modo de intra-predicción señalado para la CU.

En algunos ejemplos, un codificador de vídeo puede aplicar más de una transformación (por ejemplo, más de una transformación separable) a un valor residual para una CU. Por ejemplo, el codificador de vídeo puede transformar una TU de la CU una vez usando una primera transformación, produciendo un primer conjunto de coeficientes de transformación, a continuación, aplicando una segunda transformación para el primer conjunto de coeficientes de transformación, produciendo un segundo conjunto de coeficientes de transformación. Este proceso de aplicación de dos o más transformaciones a una TU puede denominarse como una transformación en cascada. En algunos ejemplos, la segunda transformación podrá aplicarse sólo a un subconjunto de coeficientes producidos mediante la primera transformación. Debe entenderse que la segunda transformación puede comprender una segunda transformación separable, mientras que la primera transformación puede comprender una primera transformación separable. Por lo tanto, las transformaciones en cascada pueden aplicarse mediante la aplicación de cuatro matrices totales a los coeficientes: dos para la primera transformación separable y otros dos para la segunda transformación separable.

En algunos ejemplos, la segunda transformación (es decir, la segunda transformación separable) puede corresponder a una transformación de rotación (ROT). Una transformación de rotación puede considerarse generalmente para cambiar el sistema de coordenadas de la base de transformación. Por ejemplo, un codificador de vídeo puede aplicar una primera transformación direccional, a continuación, una transformación de rotación, a una TU. Como otro ejemplo, el codificador de vídeo puede aplicar una primera DCT a una TU, a continuación, aplicar

una transformación de rotación a la TU. El codificador de vídeo puede estar configurado con varias transformaciones de rotación. El codificador de vídeo puede además estar configurado para aplicar una transformación de rotación siguiendo ciertas transformaciones direccionales y/o en combinación con ciertos modos de intra-predicción. Es decir, el codificador de vídeo puede estar configurado para aplicar una rotación de transformación para ciertas combinaciones de transformaciones direccionales y en ciertos modos de intra-predicción. Las diferentes transformaciones de rotación pueden ser indexadas mediante un valor determinado, por ejemplo, el ángulo de rotación. En algunos ejemplos, no todos los coeficientes se transforman utilizando una transformación de rotación. Por ejemplo, un codificador de vídeo puede configurarse para transformar rotacionalmente sólo coeficientes de transformación de baja frecuencia de una TU.

Además, en algunos ejemplos, las técnicas de esta divulgación incluyen el modo de intra-predicción de una TU que tiene un borde detectado dentro de la TU. Por ejemplo, una unidad de codificación de vídeo puede detectar la presencia de un borde en un bloque vecino, y luego determinar que el borde continúa en la TU actual. Los modos de predicción de gestión de bordes podrán establecerse dentro de la predicción de tal TU. El codificador de vídeo puede determinar si se debe predecir la TU utilizando el modo de predicción basada en bordes u otro modo de intra-predicción direccional. Cuando se determina que existe un borde en la TU, y cuando se selecciona el modo de predicción basada en bordes, puede usarse un valor indicativo del modo de predicción DC para indicar el modo de intra-predicción utilizado, pero debido a la determinación de la existencia del borde, este valor puede ser interpretado para indicar el modo de predicción de gestión de bordes. Además, se puede determinar el ángulo del borde y se mapea a una transformación direccional de una manera similar al mapeo de los modos de intra-predicción de dirección para las transformaciones direccionales descritas anteriormente. Del mismo modo, un patrón de exploración mapeado a la transformación direccional también puede ser utilizado en este ejemplo.

La presente divulgación también proporciona técnicas para la exploración de coeficientes de transformación para producir un vector unidimensional que luego puede ser codificado con entropía. De acuerdo con estas técnicas, un codificador de vídeo puede configurarse para seleccionar un patrón de exploración fija en base a varios factores, o para realizar una exploración adaptativa. Por ejemplo, un codificador de vídeo puede incluir un conjunto de patrones de exploración fijos. El codificador de vídeo puede seleccionar uno de los patrones de exploración fijos en base a varios criterios, tales como, por ejemplo, un modo de intra-predicción, una transformación seleccionada para una TU, si la TU se transforma usando una transformación en cascada, una transformación de rotación seleccionada para la TU, o cualquier combinación de las mismas. Por ejemplo, el codificador de vídeo puede seleccionar uno de un conjunto de análisis predefinidos sobre la base de un modo de intra-predicción, una transformación secundaria, o una combinación de los mismos. En algunos ejemplos, el codificador de vídeo puede seleccionar un índice de exploración basado en uno o más de los factores mencionados anteriormente, donde el índice de exploración puede corresponder a cualquiera de una exploración fija o adaptativa.

En algunos ejemplos, un codificador de vídeo puede configurarse para explorar de forma adaptativa coeficientes de transformación. El codificador de vídeo puede almacenar un patrón de exploración inicial fijo. A medida que el codificador de vídeo codifica los bloques de una imagen, el codificador de vídeo puede actualizar el patrón de exploración de forma adaptativa. El codificador de vídeo puede, por ejemplo, recopilar estadísticas indicativas de si los coeficientes en posiciones tienden a ser de valor cero, y si un coeficiente en una posición determinada es comúnmente con valores de cero, el codificador de vídeo puede elegir explorar ese coeficiente más tarde que otros coeficientes que comúnmente tienen valores distintos de cero. Por otra parte, el codificador de vídeo puede almacenar exploraciones separadas fijas y/o estadísticas de exploración para diversas combinaciones de factores, tales como, por ejemplo, un modo de intra-predicción, una transformación seleccionada para una TU, si la TU se transforma usando una transformación en cascada, una transformación de rotación seleccionada para la TU, o cualquier combinación de las mismas. En algunos ejemplos, un codificador de vídeo puede almacenar estadísticas separadas para cada combinación de transformaciones en cascada, por ejemplo, una primera transformación seguida por una transformación de rotación. En algunos ejemplos, el codificador de vídeo puede utilizar una exploración adaptativa cuando el codificador de vídeo aplica una transformación en cascada, y una exploración fija cuando el codificador de vídeo aplica una sola transformación.

Como se señaló anteriormente, la exploración de una matriz bidimensional de coeficientes de transformación produce un vector unidimensional que luego puede ser codificado con entropía. En algunos ejemplos, un codificador de vídeo puede codificar la entropía de los coeficientes de transformación utilizando codificación aritmética binaria adaptativa al contexto (CABAC). El codificador de vídeo también puede elementos de sintaxis que se codifican por entropía, tales como, por ejemplo, una señal de coeficiente significativo y una última señal coeficiente. Cuando se digitalizan adaptativamente los coeficientes de transformación, un codificador de vídeo puede establecer el valor de la señal de coeficiente significativo para indicar si los coeficientes son significativos o no. El codificador de vídeo, por ejemplo, puede estar configurado para determinar que un coeficiente es significativo cuando el valor del coeficiente es distinto de cero. El codificador de vídeo también puede establecer el valor de la última señal de coeficiente para indicar el último coeficiente en el vector producido por la exploración adaptativa. Un decodificador de vídeo puede utilizar estos elementos de sintaxis para actualizar las estadísticas almacenadas localmente, para explorar de manera adaptativa inversa los coeficientes de entropía codificada. Esta divulgación proporciona técnicas para la selección de un modelo de contexto cuando se realiza CABAC para codificar dichos elementos de sintaxis. El codificador de vídeo puede seleccionar el modelo de contexto basado en, por ejemplo, un modo de intra-predicción para la CU que se codifica, entre otros elementos.

La figura 1 es un diagrama de bloques que ilustra un ejemplo un sistema de codificación y de decodificación de vídeo 10 que puede utilizar técnicas de codificación y decodificación de unidades de transformación de una unidad de codificación. Como se muestra en la figura 1, el sistema 10 incluye un dispositivo de origen 12 que transmite vídeo codificado a un dispositivo de destino 14 a través de un canal de comunicación 16. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera de una amplia gama de dispositivos. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender dispositivos de comunicación inalámbricos, tales como auriculares inalámbricos, llamados radiotéfonos celulares o vía satélite, o dispositivos inalámbricos que pueden comunicar información de vídeo a través de un canal de comunicación 16, en cuyo caso el canal de comunicación 16 es inalámbrico.

Las técnicas de esta divulgación, sin embargo, que se refiere a la codificación y decodificación de unidades de transformación, no se limitan necesariamente a aplicaciones inalámbricas o ajustes. Por ejemplo, estas técnicas pueden aplicarse a emisiones de televisión en el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo de Internet, vídeo digital codificado que se codifica en un medio de almacenamiento u otros escenarios. En consecuencia, el canal de comunicación 16 puede comprender cualquier combinación de los medios inalámbricos o cableados adecuados para la transmisión o el almacenamiento de datos de vídeo codificados.

En el ejemplo de la figura 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20, un modulador/demodulador (módem) 22 y un transmisor 24. El dispositivo de destino 14 incluye un receptor 26, un módem 28, un decodificador de vídeo 30, y un dispositivo de visualización 32. De acuerdo con esta divulgación, el codificador de vídeo 20 del dispositivo de origen 12 puede estar configurado para aplicar las técnicas para la codificación y decodificación de unidades de transformación de esta divulgación. En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde una fuente de vídeo externa 18, tal como una cámara externa. Del mismo modo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

El sistema 10 ilustrado de la figura 1 es meramente un ejemplo. Las técnicas para la codificación y decodificación de unidades de transformación pueden realizarse mediante cualquier dispositivo de codificación y/o decodificación de vídeo digital. Aunque las técnicas de esta divulgación se realizan generalmente mediante un dispositivo de codificación de vídeo o un dispositivo de decodificación de vídeo, las técnicas también pueden realizarse mediante un codificador/decodificador de vídeo, típicamente llamado como un "CODEC". El dispositivo de origen 12 y el dispositivo de destino 14 son meramente ejemplos de tales dispositivos de codificación, en el que el dispositivo de origen 12 genera datos codificados de vídeo para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de una manera sustancialmente simétrica, de manera que cada uno de los dispositivos 12, 14 incluye componentes de codificación y decodificación de vídeo. Por lo tanto, el sistema 10 puede soportar la transmisión unidireccional o bidireccional de vídeo entre los dispositivos de vídeo 12, 14, por ejemplo, para transmisión de vídeo, reproducción de vídeo, difusión del vídeo, o videotelefonía.

La fuente de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado anteriormente, y/o un canal de vídeo de un proveedor de contenido de vídeo. Como otra alternativa, la fuente de vídeo 18 puede generar datos basados en gráficos computarizados como fuente de vídeo, o una combinación de vídeo en línea, 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 denominados teléfonos con cámara o teléfonos con vídeo. Como se ha mencionado anteriormente, sin embargo, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas. En cada caso, el vídeo capturado, pre-capturado, o generado por ordenador puede ser codificado por el codificador de vídeo 20. La información del vídeo codificado puede entonces modularse mediante el módem 22 de acuerdo con un estándar de comunicación, y se transmite al dispositivo de destino 14 a través del transmisor 24. El módem 22 puede incluir varios mezcladores, filtros, amplificadores u otros componentes diseñados para la modulación de la señal. El transmisor 24 puede incluir circuitos diseñados para la transmisión de datos, incluyendo amplificadores, filtros, y una o más antenas.

El receptor 26 del dispositivo de destino 14 recibe información a través del canal 16, y el módem 28 demodula la información. Una vez más, el proceso de codificación de vídeo puede implementar una o más de las técnicas descritas en la presente memoria para codificar y decodificar las unidades de transformación. La información transmitida por el canal 16 puede incluir información de la sintaxis definida por el codificador de vídeo 20, que también es utilizada por el decodificador de vídeo 30, que incluye los elementos sintácticos que describen las características y/o el procesamiento de las unidades de codificación u otras unidades de datos de vídeo codificados, por ejemplo, grupos de imágenes (GOPs), porciones, fotogramas y similares. Una estructura de datos de árbol cuádruple de CU puede formar parte de la información de sintaxis para una unidad de codificación más grande. Es decir, cada LCU puede incluir información de sintaxis en forma de un árbol cuádruple de CU, que puede describir cómo la LCU se divide en sub-CUs, así como información de señalización sobre cómo se codifican la LCU y las sub-CUs. Del mismo modo, las estructuras de datos de árbol cuádruple de TU pueden formar parte de la información de sintaxis para las CUs de hoja de la LCU, que puede describir la forma cómo las respectivas CUs de hoja se dividen

en TUs.

El decodificador de vídeo 30 puede utilizar el árbol cuádruple de CU y los árboles cuádruples de TU para determinar cómo decodificar las CUs de una imagen recibida, incluyendo las TUs de la UC. El decodificador de vídeo 30 puede entonces descifrar la CU y enviar los datos de vídeo decodificados al dispositivo de visualización 32. El dispositivo de visualización 32 muestra los datos de vídeo decodificados a un usuario, y puede comprender cualquiera de una variedad de dispositivos de visualización, tal como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCU), una pantalla de plasma, un diodo emisor de luz orgánica (OLED), u otro tipo de dispositivo de visualización.

En el ejemplo de la figura 1, el canal de comunicación 16 puede comprender cualquier medio de comunicación inalámbrico o cableado, tales como un espectro de frecuencia de radio (RF) o una o más líneas de transmisión físicas, o cualquier combinación de los medios de comunicación inalámbricos y cableados. El canal de comunicación 16 puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia, o una red global, tal como Internet. El canal de comunicación 16 representa generalmente cualquier medio adecuado de comunicación, o colección de diferentes medios de comunicación, para transmitir datos de vídeo desde el dispositivo de origen 12 al dispositivo de destino 14, incluyendo cualquier combinación adecuada de medios cableados o inalámbricos. El canal de comunicación 16 puede incluir enrutadores, conmutadores, estaciones base, o cualquier otro equipo que pueda ser útil para facilitar la comunicación del dispositivo de origen 12 al dispositivo de destino 14.

El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con un estándar de compresión de vídeo, tal como el estándar H.264 del ITU-T, denominado alternativamente como MPEG-4 Parte 10, Codificación de vídeo avanzada (AVC). Como otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden operar de acuerdo a la codificación de vídeo estándar de alta eficiencia (HEVC), y pueden ajustarse a la Prueba del Modelo HEVC (HM). Las técnicas de esta divulgación, sin embargo, no se limitan a ningún estándar de codificación particular. Otros ejemplos incluyen MPEG-2 e ITU-T H.263. Aunque no se muestra en la figura 1, en algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar cada uno integrado con un codificador y decodificador de audio, y pueden incluir unidades MUX-DEMUX apropiadas, u otro hardware y software, para gestionar la codificación de audio y vídeo en un flujo de datos común o flujos de datos independientes. Si es aplicable, las unidades MUX-DEMUX pueden cumplir con el protocolo multiplexor ITU H.223, u otros protocolos tales como el protocolo de datagramas de usuario (UDP).

El estándar ITU-T H.264/MPEG-4 (AVC) fue formulado por el Grupo de expertos de codificación de vídeo (VCEG) ITU-T junto con el Grupo de expertos de imágenes en movimiento (MPEG) ISO/IEC como el producto de una asociación colectiva conocida como el Equipo de Vídeo Conjunto (JVT). En algunos aspectos, las técnicas descritas en esta divulgación pueden ser aplicadas a dispositivos que se ajustan en general al estándar H.264. El estándar H.264 se describe en la recomendación ITU-T H.264, Codificación avanzada de vídeo para servicios audiovisuales genéricos, por el Grupo de Estudio del ITU-T, y con fecha de marzo de 2005, al que se puede hacer referencia en este documento como el estándar H.264 o la especificación H.264 o la especificación o estándar H.264/AVC. El Equipo de vídeo conjunto (JVT) continúa trabajando en extensiones H.264/MPEG-4 AVC.

El codificador de vídeo 20 y el decodificador de vídeo 30, cada uno, puede implementarse como cualquiera de una variedad de circuitos codificadores adecuados, tal como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de aplicaciones (ASIC), matrices de puertas programables (FPGAs), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan en software, un dispositivo puede almacenar las instrucciones para el software en un medio legible por ordenador adecuado no transitorio y ejecutar las instrucciones usando uno o más procesadores para llevar a cabo las técnicas de esta divulgación. Cada uno del codificador de vídeo 20 y del decodificador de vídeo 30 puede incluirse en uno o más codificadores o decodificadores, cualquiera de los cuales puede integrarse como parte de un codificador/decodificador combinado (CODEC) en una respectiva cámara, ordenador, dispositivo móvil, dispositivo de abonado, dispositivo de transmisión, caja set-top, servidor o similares.

Una secuencia de vídeo incluye típicamente una serie de fotogramas de vídeo. Un grupo de imágenes (GOP) comprende generalmente una serie de uno o más fotogramas de vídeo. Un GOP puede incluir datos de sintaxis en una cabecera del GOP, una cabecera de uno o más fotogramas del GOP, o en otros lugares, que describen una serie de fotogramas incluidos en el GOP. Cada fotograma puede incluir datos de la sintaxis del fotograma que describe un modo de codificación para el fotograma respectivo. El codificador de vídeo 20 funciona típicamente en unidades de fotogramas de vídeo individuales de codificación para codificar los datos de vídeo. Una unidad de codificación puede corresponder a una LCU o una sub-CU, y el término CU puede referirse a una LCU o a una sub-CU. La información del encabezado para una LCU puede describir el tamaño de la LCU, el número de veces que la LCU se puede dividir (referido como la profundidad CU en esta divulgación), y otra información. Cada fotograma de vídeo puede incluir una pluralidad de porciones, y cada porción puede incluir una pluralidad de LCUs.

En algunos ejemplos, la predicción se puede realizar para varios tamaños de CU. El tamaño de una LCU puede estar definido por la información de sintaxis. Suponiendo que el tamaño de una CU de hoja de nodo particular es $2N \times 2N$, los tamaños de intra-predicción pueden incluir $2N \times 2N$ o $N \times N$, en algunos ejemplos, y tamaños de inter-predicción simétricos puede incluir $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. En algunos ejemplos, la división asimétrica

puede utilizarse para inter-predicción con tamaños de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no se divide, mientras que la otra dirección se divide en un 25% y un 75%. Qué porción de la CU es la división del 25% se indica con una "n", seguido de una indicación de "Arriba", "Abajo", "Izquierda" o "Derecha". Así, por ejemplo, " $2N \times nU$ " se refiere a una CU de $2N \times 2N$ que se divide horizontalmente con una PU de $2N \times 0,5N$ en la parte superior y una PU de $2N \times 1,5N$ en la parte inferior.

En esta divulgación, " $N \times N$ " y " N por N " puede utilizarse indistintamente para referirse a las dimensiones en píxeles de un bloque (por ejemplo, CU, PU, o TU) en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de $N \times N$ generalmente tiene N píxeles en una dirección vertical y N píxeles en dirección horizontal, donde n representa un valor entero no negativo. Los píxeles de un bloque pueden estar dispuestos en filas y columnas. Por otra parte, los bloques no deben tener necesariamente el mismo número de píxeles en la dirección horizontal que en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N .

Las PU de una CU pueden comprender datos de píxeles en el dominio espacial (también conocido como el dominio de píxeles), mientras que TUs de la CU pueden transformadas para producir coeficientes en el dominio de transformación, por ejemplo, tras la aplicación de una transformación tal como una transformación discreta de coseno (DCT), una transformación de un número entero, una transformación de ondícula, o una transformación conceptualmente similar a los datos de vídeo residuales. Los datos residuales representan generalmente diferencias de píxeles entre valores de una PU y los valores de píxeles del emplazamiento común sin codificar a partir de los datos de vídeo de entrada. Los coeficientes pueden ser también cuantificados. Los coeficientes de transformaciones de la TU se puede decir que están en el dominio de la frecuencia.

El codificador de vídeo 20 puede implementar cualquiera o todas las técnicas de esta divulgación para mejorar la codificación de las unidades de transformación de una unidad de codificación. Asimismo, el decodificador de vídeo 30 puede implementar cualquiera o todas estas técnicas para mejorar la decodificación de las unidades de transformación de una unidad de codificación. En general, las técnicas de esta divulgación están dirigidas a la transformación de coeficientes de unidades de transformación que siguen al cálculo de los coeficientes basados en la predicción de modo intra. Sin embargo, ciertos aspectos de esta divulgación también se pueden implementar respecto a la codificación de inter-predicción. Para propósitos de ejemplo, estas técnicas se describen en relación con la codificación de intra-predicción de TUs. Debe entenderse que ciertos aspectos de estas técnicas también se pueden realizar en conjunción con la codificación de inter-predicción.

El codificador de vídeo 20 puede recibir una LCU y determinar si se debe dividir la LCU en cuatro cuadrantes, que comprenden, cada uno, una sub-CU, o si se debe codificar la LCU sin dividirse. A raíz de una decisión de dividir una LCU en sub-CUs, el codificador de vídeo 20 puede determinar si dividir cada sub-CU en cuatro cuadrantes, comprendiendo cada uno una sub-CU. El codificador de vídeo 20 puede continuar para determinar de forma recursiva si dividir una CU, con un número máximo de divisiones indicado por la profundidad de la LCU. El codificador de vídeo 20 puede proporcionar una estructura de datos de árbol cuádruple de CU indicativa de la división de una LCU y sub-CUs de la LCU. La LCU puede corresponder a un nodo raíz de árbol cuádruple de LCU. Cada nodo del árbol cuádruple de CU puede corresponder a una CU de la LCU. Además, cada nodo puede incluir un valor de señal de división indicativa de si se divide la CU correspondiente.

Si la LCU se divide, por ejemplo, el codificador de vídeo 20 puede fijar el valor de la señal de división en el nodo raíz para indicar que la LCU se divide. Entonces, el codificador de vídeo 20 puede fijar valores de nodos hijos del nodo raíz para indicar que, en su caso, se dividió la sub-CU de la LCU. Una CU no dividida puede corresponder a un nodo hoja de la estructura de datos de árbol cuádruple de CU, donde un nodo hoja no tiene nodos secundarios. Además, cada CU de nodo hoja puede incluir una o más TUs, como se indica mediante un árbol cuádruple de TU para la CU de nodo hoja.

El codificador de vídeo 20 puede codificar cada sub-CU de la LCU que corresponde a un nodo hoja en la estructura de datos de árbol cuádruple. Para propósitos de ejemplo, esta divulgación describe las técnicas relativas a la codificación de intra-predicción de TUs correspondientes a la CU de nodo hoja. En la codificación de modo intra, el codificador de vídeo 20 puede formar unidades de predicción (PUs) para cada TU correspondientes a un nodo hoja en la estructura de datos de la TU de árbol cuádruple. En algunos ejemplos, el codificador de vídeo 20 puede seleccionar uno de los treinta y cuatro modos de intra-predicción diferentes para la CU y la señal del modo de intra-predicción seleccionado en el nodo raíz del árbol cuádruple de TU. Comenzando con una primera TU más grande (igual en tamaño a la CU de nodo hoja en el árbol cuádruple de CU), el codificador de vídeo 20 puede determinar si dividir la mayor TU y, de forma recursiva, si se dividen sub-TUs de la TU matriz. El codificador de vídeo 20 puede señalar, además, un modo de intra-predicción en el árbol cuádruple de la CU de nodo hoja para la CU que incluyen el árbol cuádruple de TU, donde el modo de intra-predicción señalado puede describir el modo de intra-predicción que se utilizará para calcular los valores predichos para cada una de las TUs en el árbol cuádruple de TU correspondiente a la CU. El codificador de vídeo 20 recupera los datos de predicción para las TUs vecinas, los datos de vídeo codificados previamente, de acuerdo con el modo de intra-predicción seleccionado. De esta manera, las PUs de una CU predicha usando un modo de intra-predicción son del mismo tamaño que las TUs de la CU.

De acuerdo con las técnicas de la presente divulgación, si el modo de intra-predicción seleccionado no está disponible para la CU actual, por ejemplo, debido a su tamaño, el codificador de vídeo 20 puede seleccionar un modo de intra-predicción en el que se mapea el modo señalado en la raíz del árbol cuádruple. Es decir, el codificador de vídeo 20 puede incluir información que se mapea a cada uno de los modos de un gran conjunto de modos a los modos de un conjunto más pequeño, por ejemplo, un subconjunto del conjunto grande, en correspondencia de muchos a uno. El codificador de vídeo 20 puede luego intra-prededir una o más PUs para la CU utilizando el modo de intra-predicción del conjunto más pequeño. De esta manera, el codificador de vídeo 20 sólo necesita una señal de modo de intra-predicción para la LCU, aunque el codificador de vídeo 20 puede utilizar varios modos de intra-predicción de sub-CUs de la LCU sin la señalización explícita de cada uno de los modos y el sub-CU para el que se utilizan los modos. Por lo tanto, múltiples modos de intra-predicción pueden usarse sin aumentar la cantidad de información incluida en el flujo de bits, reduciendo así la sobrecarga. En otra realización, un mayor número de direcciones de predicción puede permitirse en el nivel de CU para permitir el uso el mismo modo de intra-predicción para la LCU, independientemente de los tamaños de la sub-CU o los tamaños de la PU.

El codificador de vídeo 20 también puede configurarse con modos de predicción basados en bordes para predecir TUs en una CU que el codificador de vídeo 20 determina que incluye un borde. En general, un borde corresponde a un cambio de alta frecuencia a lo largo de una línea relativamente recta a través de la TU. Por ejemplo, un borde puede producirse a lo largo del contorno de un objeto representado en la TU en contraste con un fondo también representado en la TU. Para detectar un borde en una TU, el codificador de vídeo 20 puede calcular los gradientes de píxeles en la TU y determinar la identidad del gradiente, una línea a través de la TU. Después de determinar que una TU actual incluye un borde, el codificador de vídeo 20 puede determinar si se desea utilizar el modo de predicción basado en bordes. Si se detecta un borde, y cuando se selecciona el modo de predicción basada en bordes, el codificador de vídeo 20 puede indicar el uso del modo de predicción basado en bordes utilizando un valor que de lo contrario indica el uso del modo de predicción DC. Es decir, después de detectar la presencia de un borde en un bloque actual, el codificador de vídeo 20 puede seleccionar un modo de intra-predicción a partir de un conjunto que incluye el modo de predicción basado en bordes y otros modos de predicción direccional (pero excluyendo el modo DC), y cuando se selecciona el modo de predicción basado en bordes, se señala el uso del modo de predicción basado en bordes, como si se señala el uso del modo de predicción DC.

Después de la codificación intra-predictiva o inter-predictiva para producir datos predichos por la TU de una CU, el codificador de vídeo 20 puede calcular datos residuales, que comprenden coeficientes de las TUs representativas de las diferencias píxel por píxel entre los datos predichos y los datos originales para la TU. El codificador de vídeo 20 puede formar una o más TUs incluyendo los datos residuales de la CU de esta manera. El codificador de vídeo 20 puede luego transformar las TUs. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede elegir una transformación para aplicar a una TU basada en un modo de intra-predicción utilizado para los datos de predicción de modo intra para la TU.

En algunos ejemplos, el codificador de vídeo 20 puede incluir datos de configuración que proporcionan un mapeo de muchos a uno entre un gran conjunto de modos de intra-predicción, y un conjunto más pequeño de modos de intra-predicción. Por ejemplo, el codificador de vídeo 20 puede incluir datos de configuración que proporcionan un mapeo entre los 34 modos de intra-predicción de HM y los nueve modos de intra-predicción de H.264. Por otra parte, el codificador de vídeo 20 puede incluir datos de configuración que proporcionan un mapeo entre el conjunto más pequeño de modos de intra-predicción y transformaciones direccionales. El conjunto de transformaciones direccionales puede ser del mismo tamaño que el conjunto más pequeño de modos de intra-predicción, de tal manera que hay un mapeo de uno a uno entre el conjunto más pequeño de modos de intra-predicción y el conjunto de transformaciones direccionales. De esta manera, los datos de configuración para el codificador de vídeo 20 pueden proporcionar un mapeo indirecta de muchos a uno entre el gran conjunto de modos de intra-predicción y el conjunto de transformaciones direccionales. Alternativamente, en algunos ejemplos, puede haber una correspondencia de uno a uno del gran conjunto de transformaciones direccionales para un gran conjunto de transformaciones direccionales u otras transformaciones diseñadas, tales como transformaciones discretas de coseno, transformaciones sinusoidales discretas, u otras transformaciones conceptualmente similares. En cualquier caso, utilizando el mapeo, el codificador de vídeo 20 puede elegir una transformación para cada TU basada en el modo de intra-predicción seleccionado para una CU que incluyendo la TU.

En algunos ejemplos, puede haber transformaciones direccionales múltiples posibles para una TU de un tamaño particular. En algunos ejemplos, el codificador de vídeo 20 puede indicar un modo de intra-predicción seleccionado (por ejemplo, una dirección de intra-predicción seleccionada) en la raíz de la estructura de datos de árbol cuádruple de TU correspondiente a una CU (es decir, una CU de nodo hoja en el árbol cuádruple de CU), y el modo de intra-predicción seleccionado puede aplicarse a todas las TUs de la CU. Si todas las TUs en la CU tienen tamaños de los cuales sólo uno es posible de transformar, entonces el codificador de vídeo 20 puede proceder de acuerdo con el ejemplo anterior, en el que la transformación puede derivarse de un modo de intra-predicción señalado para la LCU. Si al menos una TU en la CU es de un tamaño para el que varias transformaciones son posibles, sin embargo, entonces el codificador de vídeo 20 puede seleccionar una de las posibles transformaciones y la señal seleccionada se transforma en el nodo raíz del árbol cuádruple TU. En consecuencia, el codificador de vídeo 20 puede utilizar la transformada señalada para transformar cada TU en la CU que tiene un tamaño asociado con varias transformaciones posibles. De esta manera, el codificador de vídeo 20 puede indicar explícitamente una transformación, sin consumir un exceso de ancho de banda adicional.

En algunos ejemplos, cuando el codificador de vídeo 20 aplica un modo de intra-predicción basado en bordes para predecir un valor para una TU, el codificador de vídeo 20 puede seleccionar una transformación para aplicarse a la TU en base a un ángulo del borde. Como se describió anteriormente, el codificador de vídeo 20 puede determinar que un borde está presente en una TU actual en base a la detección de un borde en una TU vecina que comparte un límite con la TU actual. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede calcular un ángulo relativo del borde y utilizar el ángulo del borde para seleccionar una transformación direccional, de una manera similar a la selección de una transformación direccional para un modo de intra-predicción. Por ejemplo, el codificador de vídeo 20 puede comparar el ángulo del borde con ángulos para los modos de intra-predicción direccionales, determinar un modo de intra-predicción direccional que tiene un ángulo que es más cercano al ángulo del borde, y luego transformar la TU predicha de modo de predicción basado en bordes utilizando la transformación que se mapea al modo de intra-predicción determinado.

En algunos ejemplos, el codificador de vídeo 20 puede estar configurado para aplicar más de una transformación para una TU, que esta divulgación se refiere a como una transformación en cascada. Por ejemplo, la primera transformación puede corresponder a una transformación discreta de coseno (DCT) o a una transformación de Karhunen-Loève (KLT), también conocida generalmente como una transformación direccional. Cuando se selecciona una transformación direccional basada en un modo de intra-predicción mapeada a la transformación direccional, la transformación puede denominarse como un modo dependiente de transformación direccional (MDDT). Esta divulgación también se refiere a una transformación seleccionada en base a un modo de intra-predicción diseñado como una transformación, que puede incluir transformaciones direccionales, transformaciones de coseno discretas, transformaciones sinusoidales discretas, u otras transformaciones conceptualmente similares seleccionadas específicamente para un modo de predicción.

La segunda transformación puede corresponder a una transformada de rotación. En algunos ejemplos, el codificador de vídeo 20 puede estar configurado con varias transformaciones de rotación. El codificador de vídeo 20 puede seleccionar una de las transformaciones de rotación para aplicarse mediante el cálculo de los costos de distorsión de relación para cada una de las transformaciones de rotación, en algunos ejemplos. El codificador de vídeo 20 puede estar configurado para aplicar la transformación de rotación para un número más pequeño de los coeficientes que la primera transformación. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede incluir datos de configuración para las transformaciones de modo dependiente de la rotación (MDROT), incluyendo una matriz de transformación de columna y una matriz de transformación de fila. Los modos de intra-predicción pueden ser mapeados a una primera transformación, por ejemplo, una de las MDDTs, y una de las transformaciones de rotación, por ejemplo, una de las MDROTs. Por lo tanto, un modo de intra-predicción señalado para una CU también puede proporcionar una indicación de una primera transformación para aplicarse a una TU de la LCU y una segunda transformación para aplicarse a la TU. Aunque las MDROTs se describen como ejemplos, debe entenderse que la segunda transformación puede comprender otras transformaciones, tales como transformaciones de direccionales.

Mediante la transformación de coeficientes de una TU, el codificador de vídeo así produce una matriz de coeficientes de transformación. Esta matriz tiene el mismo tamaño que la TU. En general, el proceso de transformación prepara los datos residuales para la cuantificación, que comprime aún más los datos. La cuantificación se refiere generalmente a un proceso en el que los coeficientes de transformación se cuantifican para reducir posiblemente la cantidad de datos utilizados para representar los coeficientes. El proceso de cuantificación puede reducir la profundidad de los bits asociados con algunos o todos los coeficientes. Por ejemplo, un valor de n bits puede ser, redondeado a la baja, un valor de m bits durante la cuantificación, donde n es mayor que m .

En algunos ejemplos, el codificador de vídeo 20 puede utilizar una orden de exploración predefinida para explorar los coeficientes de transformación cuantificados para producir un vector que puede ser codificado por entropía. Por ejemplo, tras una transformación convencional o una transformación dependiente del modo, el codificador de vídeo 20 puede estar configurado para aplicar una exploración en zigzag. El codificador de vídeo 20 también puede estar configurado para aplicar un análisis basado en un modo de intra-predicción y/o una o más transformaciones aplicadas al bloque. En algunos ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa después de la transformación y la cuantificación de coeficientes de una TU. En algunos ejemplos, el codificador de vídeo 20 puede comprender datos de configuración que definen los diferentes esquemas de exploración para cada posible esquema de transformación. Por ejemplo, el codificador de vídeo 20 puede incluir datos de configuración que comprenden un mapeo uno a uno entre un conjunto de transformaciones direccionales y un conjunto de patrones de análisis predefinidos. Los patrones de exploración pueden ser definidos en base a pruebas empíricas de exploraciones después de una transformación direccional particular, para optimizar la colocación de los coeficientes de transformación en el vector después de la correspondiente transformación direccional. Alternativamente, el codificador de vídeo 20 puede incluir datos de configuración que definen los índices de exploración a los que se pueden mapear los modos de intra-predicción (o esquemas de transformación), donde los índices de exploración pueden indicar los análisis predefinidos o las exploraciones de adaptación.

Por consiguiente, cada transformación direccional puede tener un patrón de exploración asociado que está relativamente optimizado para esa transformación direccional basada en pruebas empíricas. Como se señaló anteriormente, el codificador de vídeo 20 no necesita señalar la transformación direccional o el patrón de exploración utilizado para una UT particular, suponiendo que exista un mapeo entre un modo de intra-predicción de una señal en

un árbol cuádruple de TU para una CU que incluyendo la TU y la transformación direccional y el patrón de exploración. En varios ejemplos, los patrones de exploración pueden ser dependientes de una primera transformación seleccionada (por ejemplo, DCT o MDDT), una segunda transformación seleccionada (por ejemplo, MDROT, DCT, u otra transformación secundaria separable), o una combinación de ambas. En algunos ejemplos, una de las dos transformaciones en cascada puede comprender una transformación diseñada aplicada en una dirección particular (por ejemplo, horizontal o vertical), y el codificador de vídeo 20 puede seleccionar una orden de exploración que corresponde generalmente a la misma dirección o en una dirección ortogonal, en base a los datos de configuración.

En los ejemplos donde el codificador de vídeo 20 aplica una transformación en cascada a una TU, el codificador de vídeo 20 puede explorar coeficientes de manera adaptativa resultantes de la transformación en cascada. Para realizar una exploración adaptativa, el codificador de vídeo 20 puede generalmente seguir las estadísticas indicativas de si una posición particular en la matriz de coeficientes de transformaciones es más o menos propensa a ser significativa (por ejemplo, distinta de cero). El codificador de vídeo de 20 podrá adaptar el patrón de exploración a lo largo del tiempo, de tal manera que el patrón de exploración corresponde a estas probabilidades estadísticas. Es decir, el patrón de exploración adaptativa puede tratar de asegurarse que los coeficientes de transformación que tiene una probabilidad relativamente mayor de ser significativa (por ejemplo, distinta de cero) se exploran antes que los coeficientes de transformación que tienen una probabilidad relativamente baja de ser significativos. Alternativamente, el codificador de vídeo 20 puede seleccionar un índice de exploración para que la transformación en cascada sea mapeada.

El codificador de vídeo 20 puede seguir las estadísticas de exploración para cada posible transformación en cascada por separado. Por ejemplo, la probabilidad de que una posición de coeficiente particular en la matriz de transformaciones pueda variar en función de la primera y segunda transformaciones aplicadas durante la transformación en cascada. Por lo tanto, el codificador de vídeo 20 puede seguir conjuntos separados, independientes de las estadísticas para cada posible transformación en cascada. Como un ejemplo, suponiendo que los modos de intra-predicción se mapean a la vez a una MDDT y a una MDROT (u otra transformación separable secundaria), el codificador de vídeo 20 puede seguir las estadísticas independientes para cada combinación de MDDT y MDROT (u otra transformación secundaria) aplicadas a TUs. Como otro ejemplo, el codificador de vídeo 20 puede estar configurado para realizar la transformación en cascada sólo cuando el codificador de vídeo 20 aplica una DCT a una TU. Por lo tanto, el codificador de vídeo 20 puede realizar una exploración adaptativa, y seguir estadísticas independientes para realizar la exploración adaptativa, en base a una MDROT seleccionada (u otra transformación secundaria separable) para la TU aplicada siguiendo la DCT.

En algunos ejemplos, usando una exploración adaptativa o una exploración predeterminada, el codificador de vídeo 20 puede ajustar a cero los coeficientes de la matriz después de la exploración. Es decir, el codificador de vídeo 20 puede establecer valores de coeficientes en posiciones N hasta el final de la matriz, iguales a cero. El valor de N puede relacionarse con el tamaño de la CU y/o con el tamaño de la TU. En algunos ejemplos, el codificador de vídeo 20 puede ajustar a cero los coeficientes de transformación en la matriz antes de la exploración, por ejemplo, todos los coeficientes en la matriz distintos de los coeficientes en la esquina superior izquierda de la matriz.

Después de explorar la matriz de transformación para formar un vector unidimensional, el codificador de vídeo 20 puede codificar por entropía el vector unidimensional, por ejemplo, de acuerdo con el contenido adaptativo de codificación de longitud variable (CAVLC), el contexto adaptativo de codificación aritmética binaria (CABAC), en base a la codificación aritmética binaria adaptativa de contexto basada en la sintaxis (SBAC), u otra metodología de codificación de entropía.

Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en VLC pueden construirse de tal manera que los códigos relativamente más cortos corresponden a los símbolos más probables, mientras que los códigos más largos corresponden a símbolos menos probables. De esta manera, el uso o VLC pueden lograr un ahorro de bits sobre, por ejemplo, el uso de palabras de código de igual longitud para cada símbolo a transmitir.

Para realizar la CABAC, el codificador de vídeo 20 puede seleccionar un modelo de contexto para aplicar a un contexto determinado para codificar los símbolos a transmitir. El contexto puede estar relacionado, por ejemplo, si los valores vecinos son diferentes de cero o no. El codificador de vídeo 20 puede también codificar por entropía elementos de sintaxis, tales como una señal de coeficiente significativa y una última señal de coeficiente producida cuando se realiza una exploración adaptativa. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede elegir el modelo de contexto utilizado para codificar estos elementos sintácticos basados en, por ejemplo, una dirección de intra-predicción, una posición de exploración del coeficiente correspondiente a los elementos de sintaxis, el tipo de bloque, y/o el tipo de transformación, entre otros factores utilizados para la selección del modelo de contexto.

El decodificador de vídeo 30 puede operar de una manera esencialmente simétrica a la del codificador de vídeo 20. Por ejemplo, el decodificador de vídeo 30 puede recibir datos codificados de entropía representativos de una CU codificada, incluyendo los datos codificados de la TU. Estos datos recibidos pueden incluir información indicativa de un modo de intra-predicción utilizado para codificar los datos de la PU, suponiendo que la CU fue codificada con

intra-predicción. El decodificador de vídeo 30 puede invertir la codificación de entropía de los datos recibidos, formando coeficientes de cuantificación codificados. Cuando el codificador de vídeo 20 codifica por entropía los datos usando un algoritmo de código de longitud variable, el decodificador de vídeo 30 puede usar una o más tablas de VLC para determinar un símbolo correspondiente a una palabra de código recibida. Cuando el codificador de vídeo 20 codifica por entropía los datos utilizando una aritmética de acuerdo con el algoritmo, el decodificador de vídeo 30 puede utilizar un modelo de contexto para decodificar los datos, que pueden corresponder a un mismo modelo de contexto utilizado por el codificador de vídeo 20 para codificar los datos.

El decodificador de vídeo 30 puede entonces invertir la exploración de los coeficientes decodificados usando una exploración inversa que refleja la exploración utilizada por el codificador de vídeo 20. Para invertir la exploración de manera adaptativa de los coeficientes, el decodificador de vídeo 30 puede decodificar los elementos de sintaxis incluidos en la señal de coeficientes significativos y las últimas señales de coeficientes para regenerar las estadísticas utilizadas por el codificador de vídeo 20 para realizar la exploración adaptativa. El decodificador de vídeo 30 de ese modo puede formar una matriz de dos dimensiones, a partir del vector unidimensional resultante del proceso de decodificación de entropía.

A continuación, el decodificador de vídeo 30 puede invertir la cuantificación de los coeficientes en la matriz de dos dimensiones producida por la exploración inversa. El decodificador de vídeo 30 puede entonces aplicar una o más transformaciones inversas para la matriz de dos dimensiones. Las transformaciones inversas pueden corresponder a las transformaciones aplicadas por el codificador de vídeo 20. El decodificador de vídeo 30 puede determinar las transformaciones inversas para aplicarse sobre la base de, por ejemplo, el modo de intra-predicción utilizado para calcular los coeficientes para la TU, y si varias transformaciones están disponibles para una TU de un tamaño particular, la información señalada en la raíz de un árbol cuádruple de TU correspondiente a la CU que incluye la TU que se está decodificando actualmente. De esta manera, el decodificador de vídeo 30 puede seleccionar una o más transformaciones inversas para aplicarse para invertir coeficientes cuantificados para una TU para reproducir la TU, en base a un modo de intra-predicción para la TU. Por otra parte, el decodificador de vídeo 30 puede calcular un valor predicho para la TU usando un modo de intra-predicción correspondiente a una indicación señalada del modo de intra-predicción, por ejemplo, en el árbol cuádruple de TU. En algunos ejemplos, el decodificador de vídeo 30 puede determinar que un borde está presente en la TU, en base a un borde detectado en una TU vecina y, cuando se indica un modo DC, en lugar de utilizar un modo basado en bordes para predecir un valor para la TU.

El codificador de vídeo 20 y el decodificador de vídeo 30, cada uno, pueden implementarse como cualquiera de una variedad de circuitos codificadores o decodificadores adecuados, como corresponda, tal como uno o más microprocesadores, procesadores de señales digitales (DSPs), circuitos integrados de aplicación específica (ASICs), matrices de campo de puertas programables (FPGAs), circuitos lógicos discretos, software, hardware, firmware o cualquier combinación de los mismos. Cada uno del codificador de vídeo 20 y del decodificador de vídeo 30 puede incluirse en uno o más codificadores o decodificadores, cualquiera de los cuales puede integrarse como parte de un codificador/decodificador de vídeo combinado (CODEC). Un aparato que incluye el codificador de vídeo 20 y/o el decodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador, y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

La figura 2 es un diagrama de bloques que ilustra un ejemplo del codificador de vídeo 20 que puede implementar cualquiera o todas las técnicas para la codificación de unidades de transformación de datos de vídeo descritos en esta divulgación. El codificador de vídeo 20 puede realizar intra- e inter-codificación de las CUs en los fotogramas de vídeo. La infra-codificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en vídeo dentro de un fotograma de vídeo dado. La inter-codificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal entre un fotograma actual y los fotogramas codificados previamente de una secuencia de vídeo. El modo intra (modo I) puede referirse a cualquiera de los diversos modos de compresión de base espacial y modos inter, tales como la predicción de unidad direccional (modo P) o la predicción bidireccional (modo B) puede referirse a cualquiera de varios modos de compresión de base temporal.

Como se muestra en la figura 2, el codificador de vídeo 20 recibe un bloque de vídeo actual en un fotograma de vídeo a codificar. En el ejemplo de la figura 2, el codificador de vídeo 20 incluye la unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de intra-predicción 46, una memoria de fotogramas de referencia 64, un sumador 50, una unidad de transformación 52, una unidad de cuantificación 54, y una unidad de codificación de entropía 56. La unidad de transformación 52 que se ilustra en la figura 2 es la unidad que realiza la transformación real, que no debe confundirse con una TU, de una CU. Para el bloque de reconstrucción de vídeo, el codificador de vídeo 20 también incluye una unidad de cuantificación inversa 58, una unidad de transformación inversa 60, y un sumador 62. Además, el codificador de vídeo 20 puede incluir datos de configuración, como mapeo de datos 66. Un filtro de desbloqueo (no mostrado en la figura 2) también se puede incluir para filtrar los límites del bloque para eliminar artefactos de formación de bloques de vídeo reconstruidos. Si se desea, el filtro de desbloqueo típicamente filtra la salida del sumador 62.

Durante el proceso de codificación, el codificador de vídeo 20 recibe un fotograma de vídeo o una porción a codificar. El fotograma o porción pueden dividirse en múltiples bloques de vídeo, por ejemplo, unidades de codificación más grandes (LCUs). La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 realizan la codificación inter-predictiva del bloque de vídeo recibido con relación a uno o más bloques

en uno o más fotogramas de referencia para proporcionar la compresión temporal. La unidad de intra-predicción 46 puede llevar a cabo la codificación intra-predictiva del bloque de vídeo recibido respecto a uno o más bloques vecinos en el mismo fotograma o porción como el bloque a codificar para proporcionar compresión espacial.

La unidad de selección de modo 40 puede seleccionar uno de los modos de codificación, intra o inter, por ejemplo, con base a los resultados de error (a veces conocido como distorsión), y proporciona el bloque intra- o inter-codificado resultante al sumador 50 para generar datos de bloques residuales y al sumador 62 para reconstruir el bloque codificado para su uso en un fotograma de referencia. Algunos fotogramas de vídeo pueden ser designados fotogramas I, donde todos los bloques en un fotograma I se codifican en un modo de intra-predicción. En algunos casos, la unidad de intra-predicción 46 puede realizar la codificación de intra-predicción de un bloque en un fotograma P o B, por ejemplo, cuando la búsqueda por movimiento realizada por la unidad de estimación de movimiento 42 no resulta en una predicción suficiente del bloque.

La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar muy integrados, pero se ilustran por separado para fines conceptuales. La estimación de movimiento es el proceso de generación de vectores de movimiento que estiman el movimiento de bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una unidad de predicción en un fotograma actual en relación con una muestra de referencia de un fotograma de referencia. Una muestra de referencia es un bloque que se encuentra para asemejarse a la porción de la CU que incluye la PU que está codificada en términos de diferencia de píxeles, que puede determinarse por la suma de las diferencias absolutas (SAD), la suma de las diferencias cuadradas (SSD), u otras métricas de diferencia. La compensación de movimiento, realizada por la unidad de compensación de movimiento 44, puede implicar ir a buscar o generar valores para la unidad de predicción en base al vector de movimiento determinado por la estimación de movimiento. Una vez más, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar funcionalmente integradas, en algunos ejemplos.

La unidad de estimación de movimiento 42 calcula un vector de movimiento para una unidad de predicción de un fotograma inter-codificado mediante la comparación de la unidad de predicción para hacer referencia a las muestras de un fotograma de referencia almacenado en la memoria de fotogramas de referencia 64. En algunos ejemplos, el codificador de vídeo 20 podrá calcular valores de posiciones de píxeles sub-enteros de fotogramas de referencia almacenados en la memoria de fotogramas de referencia 64. Por ejemplo, el codificador de vídeo 20 puede calcular los valores de las posiciones de píxel de un cuarto, las posiciones de píxel de un octavo, y otras posiciones de píxel fraccionales del fotograma de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento respecto a las posiciones de píxeles completas y las posiciones de píxeles fraccionarios y la salida de un vector de movimiento con una precisión de píxel fraccionario. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación de entropía 56 y la unidad de compensación de movimiento 44. La porción del fotograma de referencia identificado por un vector de movimiento puede ser denominada como una muestra de referencia. La unidad de compensación de movimiento 44 puede calcular un valor de predicción para una unidad de predicción de una CU actual, por ejemplo, mediante la recuperación de la muestra de referencia identificada por un vector de movimiento para la PU.

La unidad de intra-predicción 46 puede codificar mediante intra-predicción el bloque recibido, como alternativa a la inter-predicción realizada por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44. La unidad de intra-predicción 46 puede codificar el bloque recibido respecto al bloque vecino previamente codificado, por ejemplo, los bloques de arriba, encima y a la derecha, arriba y a la izquierda, o hacia la izquierda del bloque actual, asumiendo una orden de codificación de izquierda a derecha, de arriba a abajo para los bloques (tal como una orden de exploración de fotogramas). La unidad de intra-predicción 46 se puede configurar con una variedad de diferentes modos de intra-predicción. Por ejemplo, la unidad de intra-predicción 46 puede estar configurada con un cierto número de modos de predicción direccionales, por ejemplo, 34 modos de predicción direccionales, basados en el tamaño de la CU que está siendo codificada.

La unidad de intra-predicción 46 puede seleccionar un modo de intra-predicción, por ejemplo, el cálculo de los valores de error para los diversos modos de intra-predicción y la selección de un modo que produce el valor de error más bajo. Los modos de predicción direccionales pueden incluir funciones para combinar valores de los píxeles espacialmente vecinos y la aplicación de los valores combinados de una o más posiciones de los píxeles en una PU. Una vez que se han calculado los valores para todas las posiciones de los píxeles de la PU, la unidad de intra-predicción 46 puede calcular un valor de error para el modo de predicción basado en diferencias de píxeles entre la PU y el bloque recibido a codificar. La unidad de intra-predicción 46 puede seguir probando modos de intra-predicción hasta que se descubre un modo de intra-predicción del que se obtiene un valor de error aceptable. La unidad de intra-predicción 46 puede entonces enviar la PU al sumador 50.

De acuerdo con las técnicas de esta divulgación, la unidad de intra-predicción 46 puede estar configurada para predecir un bloque que incluye un borde utilizando un modo de predicción basado en bordes. En particular, la unidad de intra-predicción 46 puede analizar los píxeles de los bloques vecinos previamente codificados para determinar si se detecta un borde en al menos uno de los bloques vecinos, y si el borde cruza una frontera entre el bloque previamente codificado y el bloque actual. Para detectar el borde, la unidad de intra-predicción 46 puede calcular los gradientes de los píxeles en los bloques vecinos codificados previamente en las direcciones horizontal y vertical. Cuando los gradientes para una pluralidad de píxeles en uno de los bloques vecinos son relativamente

perpendiculares a una línea común que cruza una frontera entre el bloque vecino y el bloque actual, la unidad de intra-predicción 46 puede determinar que el bloque actual también incluye un borde (en particular, un borde largo de la línea detectada como se ha descrito anteriormente). Debe entenderse que el término "borde" en este contexto se refiere a un cambio de alta frecuencia a lo largo de una línea relativamente recta dentro de un bloque de píxeles, y no a la frontera o límite entre bloques codificados por separado.

Por consiguiente, cuando se detecta un borde en el bloque, la unidad de intra-predicción 46 puede determinar si se predice el bloque mediante un modo de predicción basado en bordes o un modo de infra-predicción direccional. Cuando la unidad de intra-predicción 46 selecciona el modo de predicción basado en bordes, la unidad de intra-predicción 46 puede señalar que el modo de predicción DC se utilizó para predecir el bloque, con el fin de evitar el aumento del número de valores necesarios para indicar los modos de intra-predicción. Esto, como se describe en mayor detalle respecto a la figura 3, un decodificador de vídeo tal como decodificador de vídeo 30 puede configurarse para interpretar una indicación (por ejemplo, una señal o información de sintaxis) que el modo de predicción DC se utilizó para predecir un bloque como una indicación de un modo de predicción basado en bordes cuando el decodificador de vídeo determina que un borde está presente en el bloque.

El Codificador de vídeo 20 forma un bloque residual restando los datos de predicción calculados por la unidad de compensación de movimiento 44 o la unidad de infra-predicción 46 del bloque de vídeo original que se está codificando. El sumador 50 representa el componente o componentes que realizan esta operación de resta. El bloque residual puede corresponder a una matriz bidimensional de valores, donde el número de valores en el bloque residual es el mismo que el número de píxeles de la PU correspondientes al bloque residual. Los valores en el bloque residual pueden corresponder a las diferencias entre los píxeles colocados en la PU y en el bloque original a codificar.

La unidad de transformación 52 puede formar una o más unidades de transformación (TUs) a partir del bloque residual. La unidad de transformación 52 aplica una transformación, tal como una transformación discreta de coseno (DCT), una transformación direccional, o una transformación conceptualmente similar, a la TU, produciendo un bloque de vídeo que comprende coeficientes de transformación. De acuerdo con las técnicas de esta divulgación, la unidad de infra-predicción 46 puede enviar una indicación del modo de intra-predicción seleccionado a la TU a la unidad de transformación 52, por ejemplo, mediante el modo de señalización en un nodo o un árbol cuádruple de TU correspondiente a la TU. En consecuencia, la unidad de transformación 52 puede seleccionar una transformación para aplicarse a la TU basada en la indicación del modo de intra-predicción recibido desde la unidad de intra-predicción 46.

En algunos ejemplos, la unidad de transformación 52 está configurada para seleccionar unas transformaciones, tales como una transformación direccional para aplicar a la TU basada en un modo de intra-predicción utilizado para predecir la TU. Es decir, los datos de mapeo 66 pueden almacenar los datos de configuración que describen una transformación que se aplicará sobre la base del modo de intra-predicción utilizado para predecir la TU. De esta manera, la unidad de transformación 52 puede transformar un bloque residual usando una transformación mapeada a un modo de intra-predicción utilizado para calcular el bloque residual. Asimismo, los datos de mapeo 66 pueden mapear un modo de intra-predicción, una transformación, o ambos, a un índice de exploración en particular, que puede ser utilizado para seleccionar una exploración a aplicarse a coeficientes de transformación cuantificados del bloque.

En algunos ejemplos, los datos de mapeo 66 pueden almacenar los datos de configuración que incluyen un mapeo de muchos a uno entre un gran conjunto de modos de intra-predicción y un pequeño conjunto de modos de intra-predicción. El conjunto más pequeño puede comprender un subconjunto de los modos de intra-predicción. En algunos ejemplos, el gran conjunto puede comprender modos de intra-predicción soportados por el modelo de prueba HEVC, mientras que el conjunto más pequeño puede comprender modos de intra-predicción soportados por H.264. Los datos de mapeo 66 también pueden incluir un mapeo, tal como un mapeo de uno a uno, entre el más pequeño conjunto de modos de intra-predicción y la transformación. El conjunto más pequeño de los modos de intra-predicción puede ser un mapeo de uno a uno a transformaciones direccionales que están diseñadas para proporcionar los resultados de la transformación empíricamente mejores para el modo de intra-predicción correspondiente.

En algunos ejemplos, los modos de intra-predicción del gran conjunto y del conjunto más pequeño pueden estar asociados con respectivos ángulos de predicción. Por tanto, el mapeo entre el gran conjunto de modos de predicción y el conjunto más pequeño puede determinarse mediante la selección de un ángulo de predicción de uno de los modos de intra-predicción en el conjunto más pequeño que se aproxime al ángulo de predicción de un modo de intra-predicción del gran conjunto. Supongamos que α representa el ángulo de predicción de uno de los modos de intra-predicción del gran conjunto, y que β_i representa el ángulo de predicción del i -ésimo modo de intra-predicción del conjunto más pequeño. Para mapear el modo de intra-predicción a uno de los modos de intra-predicción del conjunto más pequeño, el codificador de vídeo 20 puede, dado α , encontrar β_i de manera que se cumple la ecuación 1 siguiente:

$$\arg_{\{\beta_i\}} \min (\min(\text{abs}(\alpha - \beta_i), \text{abs}(-\alpha - \beta_i))) \quad (1)$$

La unidad de transformación 52 puede recibir la indicación del modo de intra-predicción seleccionado desde la unidad de intra-predicción 46, a continuación, consultar los datos de mapeo 66 para determinar una transformación a utilizar para transformar una TU incluyendo los valores residuales recibidos del sumador 50. Si la TU es de un tamaño para los que son posibles varias transformaciones, la unidad de transformación 52 u otra unidad de codificador de vídeo 20 puede seleccionar una transformación para la TU de ese tamaño, de tal manera que la unidad de transformación 52 puede aplicar la misma transformación para todas las TUs de ese tamaño en la misma CU. En algunos ejemplos, la unidad de transformación 52 puede además estar configurada para realizar una transformación de rotación en cascada a partir de la primera transformación. Es decir, después de la primera transformación, la unidad de transformación 52 puede seleccionar y aplicar una transformación de rotación para los coeficientes de transformación. La unidad de transformación 52 puede seleccionar la transformación de rotación basada en, por ejemplo, el modo de intra-predicción utilizado para predecir la PU para la TU actual.

Como se describió anteriormente, en algunos ejemplos, la unidad de intra-predicción 46 está configurada para determinar si un bloque incluye un borde (es decir, un cambio de alta frecuencia entre los píxeles dentro del bloque). Cuando se detecta un borde, la unidad de intra-predicción 46 puede seleccionar un modo de predicción basado en bordes, o un modo de intra-predicción direccional convencional. Es decir, la unidad de intra-predicción 46 puede sustituir el modo de predicción DC por el modo de predicción basado en bordes. En algunos ejemplos, cuando el bloque se predijo usando el modo de predicción basado en bordes, la unidad de transformación 52 selecciona una transformación (tal como una transformación direccional) mapeada a un modo de intra-predicción direccional que tiene un ángulo que se aproxima mucho a un ángulo del borde. Es decir, la unidad de transformación 52 en algunos ejemplos determina un ángulo del borde dentro del bloque y selecciona una transformación que se mapea a un modo de intra-predicción direccional que tiene un ángulo que se aproxima estrechamente (por ejemplo, tiene una distancia mínima relativa a) el ángulo del borde. En algunos ejemplos, la unidad de transformación 52 está configurada para seleccionar una transformación que se mapea a un modo de intra-predicción direccional que tiene un ángulo que se aproxima estrechamente al ángulo de un borde detectado en un bloque.

El codificador de vídeo 20 también puede indicar el uso del modo de predicción basado en bordes utilizando un valor que de otro modo se utiliza para señalar el uso del modo de predicción DC. Por lo tanto, aunque el modo de predicción DC se señala, el modo de predicción basado en bordes puede ser usado para predecir el bloque. Asimismo, aunque el modo de predicción basado en bordes puede ser usado para predecir el bloque, la unidad de transformación 52 puede utilizar la transformación mapeada al modo de intra-predicción que tiene un ángulo que se aproxima al ángulo del borde detectado en el bloque.

Los datos de mapeo 66 pueden proporcionar datos de configuración que indican que cuando se detecta un borde en un bloque para el que se indica el modo de predicción DC, la unidad de transformación 52 selecciona una transformación que tiene un ángulo que se aproxima al ángulo del borde. Además, como se describió anteriormente, los datos de mapeo 66 pueden incluir un mapeo de los modos de intra-predicción y los ángulos para los modos de intra-predicción (que pueden definir los modos de intra-predicción) para transformaciones direccionales. En consecuencia, la unidad de transformación 52 puede consultar los datos de mapeo 66 para determinar un modo de intra-predicción que tiene un ángulo que se aproxima estrechamente al ángulo de un borde en un bloque, así como para determinar la transformación que se mapea al modo de intra-predicción determinado.

De esta manera, el codificador de vídeo 20 es un ejemplo de un codificador de vídeo configurado para calcular un bloque residual para un bloque de datos de vídeo basada en una forma de bloque predicha utilizando un modo de intra-predicción, y transformar el bloque residual usando una transformación de mapeo en el modo de intra-predicción. El codificador de vídeo 20 es también un ejemplo de un codificador de vídeo configurado para recibir una indicación, desde un primer modo de intra-predicción en un primer conjunto de modos de intra-predicción para un bloque de datos de vídeo, determinar un segundo modo de intra-predicción a partir de un conjunto más pequeño de modos de intra-predicción al que se mapea el primer modo de intra-predicción, determinar una transformación direccional a la que se mapea el segundo modo de intra-predicción, y aplicar la transformación direccional a los datos residuales del bloque.

Además, el codificador de vídeo 20 es también un ejemplo de un codificador de vídeo configurado para seleccionar un modo de intra-predicción a utilizar para codificar un bloque de datos de vídeo, determinar si el bloque incluye sub-bloques de un tamaño para el que varias transformaciones son posibles en función del tamaño del sub-bloque y el modo de intra-predicción seleccionado, cuando el bloque incluye el sub-bloque del tamaño para que varias transformaciones sean posibles en función del tamaño del sub-bloque y el modo de intra-predicción seleccionado, seleccionar una de las varias transformaciones posibles, transformar el sub-bloque con la seleccionada de las posibles varias transformaciones., y proporcionar una indicación de la seleccionada de las varias transformaciones posibles para el tamaño del bloque.

Por otra parte, el codificador de vídeo 20 es un ejemplo de un codificador de vídeo configurado para determinar que un bloque codificado con intra-predicción contiene un borde dentro del bloque, calcular un bloque residual para el bloque basado en un valor de predicción calculado utilizando el modo de intra-predicción dirigido a bordes, transformar el bloque residual utilizando una transformación direccional mapeada a un modo de intra-predicción direccional que tiene un ángulo que se aproxima a un ángulo del borde, y enviar información representativa del bloque residual transformado e información que indica que el bloque se predijo utilizando un modo de intra-

predicción DC.

La unidad de transformación 52 puede enviar los coeficientes de transformación resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 puede entonces cuantificar los coeficientes de transformación. En algunos ejemplos, la unidad de cuantificación 54 puede entonces realizar una exploración de la matriz que incluye los coeficientes de transformación cuantificados. Alternativamente, la unidad de codificación de entropía 56 puede realizar la exploración. Esta divulgación describe la unidad de codificación de entropía 56 como la realización de la exploración, aunque debe entenderse que, en otros ejemplos, otras unidades de procesamiento, tales como la unidad de cuantificación 54, pueden realizar la exploración.

En algunos ejemplos, la unidad de codificación de entropía 56 puede recibir una indicación del modo de intra-predicción seleccionado dentro de la unidad de predicción 46 o de la unidad de transformación 52. La unidad de codificación de entropía 56 puede seleccionar una exploración para aplicarse a la matriz de coeficientes de transformación, para convertir la matriz de dos dimensiones en un vector unidimensional. En algunos ejemplos, la unidad de codificación de entropía 56 selecciona una exploración de un conjunto predeterminado de las exploraciones. Los datos del mapeo 66 pueden mapear el más pequeño conjunto de modos de intra-predicción al conjunto predeterminado de exploraciones. La unidad de codificación de entropía 56 puede seleccionar la exploración basada en varias características de la TU actual, tales como, por ejemplo, el tipo de bloque (intra o inter), el modo de intra-predicción (suponiendo un bloque intra-codificado), y/o un tipo de transformación aplicada a la TU (por ejemplo, DCT o KLT).

En algunos ejemplos, la unidad de codificación de entropía 56 puede estar configurada para realizar una exploración adaptativa. Inicialmente (por ejemplo, para una primera TU de una fotograma actual), la unidad de codificación de entropía 56 puede utilizar un patrón de exploración predeterminado. Con el tiempo, la unidad de codificación de entropía 56 puede actualizar el patrón de exploración para realizar la exploración de adaptación. En general, el objetivo de la exploración adaptativa es determinar una probabilidad de que un determinado coeficiente de transformación será distinto de cero. A continuación, el orden de exploración en general procede de coeficientes con la más alta probabilidad de ser distintos de cero a la menor probabilidad de ser distintos de cero. La unidad de codificación de entropía 56 puede determinar estas probabilidades a lo largo del tiempo utilizando diversas estadísticas y cálculos. Por otra parte, la unidad de codificación de entropía 56 puede realizar un seguimiento de estadísticas separadas para cada modo de intra-predicción, transformación, transformación en cascada, o cualquier combinación de los mismos.

La unidad de codificación de entropía 56 puede utilizar una tabla de alto rango dinámico y actualizar dinámicamente tablas de exploración para determinar las probabilidades de coeficientes de transformación que son distintos de cero, y para determinar el orden de exploración adaptativa. Suponiendo una TU $N \times N$, cada una de estas tablas pueden ser tablas $N \times N$ con los valores correspondientes a los coeficientes de transformación de la TU. La tabla de alto rango dinámico puede ser una tabla predeterminada fija que proporciona probabilidades que cada coeficiente de transformación no sea cero. Esta tabla se puede calcular sobre la base de un conjunto de datos de entrenamiento. Por otra parte, esta tabla puede ser utilizada para proporcionar el punto de partida para la orden de exploración adaptativa.

La unidad de codificación de entropía 56 puede actualizar la tabla de exploración dinámicamente actualizada a lo largo del tiempo para reflejar estadísticas recientemente determinadas de coeficientes de transformación. La unidad de codificación 56 puede mantener un recuento del número de veces en las que cada coeficiente en una posición particular en la matriz de transformaciones de entropía $N \times N$ es distinto de cero. Es decir, para cada TU de un fotograma actual, la unidad de codificación de entropía 56 puede incrementar los valores en la tabla de exploración dinámicamente actualizada correspondiente a los coeficientes distintos de cero en el bloque de transformación actual, concretamente, en la tabla de exploraciones dinámicamente actualizada asociada con el modo de intra-predicción, transformaciones, y/o transformaciones en cascada para CU actual. Por ejemplo, si el coeficiente de transformación en la fila 2 y la columna 1 no es cero, la unidad de codificación de entropía 56 puede añadir uno al valor en la tabla de exploración de actualización dinámica en la fila 2 y la columna 1. La unidad de codificación de entropía 56 también puede normalizar periódicamente los valores de la tabla de exploración de actualización dinámica, para evitar los valores que exceden de un valor máximo.

Para realizar la exploración adaptativa para una primera TU de una fotograma actual, la unidad de codificación de entropía 56 puede explorar en base a la tabla de alto rango dinámico solamente. La unidad de codificación de entropía 56 también puede inicializar la tabla de exploración de actualización dinámica mediante, por ejemplo, el establecimiento de todos los valores en la tabla de exploración de actualización dinámica a cero. Para cada coeficiente diferente de cero en el bloque de transformación, la unidad de codificación de entropía 56 puede añadir uno al valor mapeado en la tabla de exploración de actualización dinámica asociada con el modo de intra-predicción y transformación o transformación en cascada para la TU actual. Para la TU posterior se usa el mismo modo de intra-predicción y transformación o transformación en cascada, la unidad de codificación de entropía 56 puede referirse primero a la tabla de exploración de actualización dinámica para determinar cuál de los coeficientes de transformación es más probable que sea distinto de cero, a continuación, explorar para disminuir la probabilidad de los coeficientes sean distintos de cero. En algunos casos, dos o más valores de la tabla de exploración de actualización dinámica pueden ser iguales. En tal caso, la unidad de codificación de entropía 56 se refiere a la tabla

dinámica de alto rango para determinar cuál de los coeficientes se explora a continuación. De esta manera, la unidad de codificación de entropía 56 puede realizar una exploración adaptativa para cada modo de intra-predicción, transformación o transformación en cascada (o cualquier combinación de los mismos), en base a una combinación de una tabla dinámica de alto rango y una tabla de exploración de actualización dinámica.

- 5 Mediante la exploración de la matriz bidimensional de coeficientes de transformación, la unidad de codificación de entropía 56 puede producir una matriz unidimensional que incluye los coeficientes de transformación. La unidad de codificación de entropía 56 puede entonces analizar la TU para formar una matriz y cuantificar los coeficientes de transformación de la matriz después de la exploración para reducir aún más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociados con algunos o todos de los coeficientes. El grado de
- 10 cuantificación puede codificarse mediante el ajuste de un parámetro de cuantificación.

La unidad de codificación de entropía 56 puede también codificar por entropía elementos de sintaxis para los coeficientes de la matriz antes de o durante la exploración adaptativa. Los elementos de sintaxis pueden incluir una señal de coeficiente significativo que indica si un coeficiente particular es significativo (por ejemplo, distinto de cero) y una última señal de coeficiente que indica si un coeficiente particular es el último coeficiente explorado en la exploración adaptativa. Un decodificador de vídeo puede utilizar estos elementos de sintaxis para reconstruir la tabla de exploración de actualización dinámica, de tal manera que el decodificador de vídeo puede explorar a la inversa los coeficientes codificados por unidad de codificación de entropía 56.

15

Para codificar por entropía los elementos de sintaxis, la unidad de codificación de entropía 56 puede realizar CABAC y seleccionar algunos modelos de contexto en base a, por ejemplo, el número de coeficientes significativos en los coeficientes previamente escaneados N, donde N es un valor entero que puede estar relacionado con el tamaño del bloque que se está explorando. La unidad de codificación de entropía 56 también puede seleccionar el modelo de contexto basado en un modo de predicción utilizado para calcular los datos residuales que se transformaron en el bloque de coeficientes de transformación, y un tipo de transformación utilizada para transformar los datos residuales en el bloque de coeficientes de transformación. Cuando los datos de predicción correspondientes se predijeron usando un modo de intra-predicción, la unidad de codificación de entropía 56 puede basar aún más la selección del modelo de contexto en la dirección del modo de infra-predicción.

20

25

De esta manera, el codificador de vídeo 20 representa un ejemplo de un codificador de vídeo configurado para transformar los datos para un bloque residual, de datos de vídeo mediante una primera transformación para producir un bloque intermedio bidimensional de coeficientes de transformación, transformar el bloque intermedio bidimensional de coeficientes de transformación usando una transformación de rotación para producir un bloque bidimensional de coeficientes de transformación, seleccionar un conjunto de estadísticas asociadas con al menos una de la primera transformación y la transformación de rotación, en el que el conjunto de estadísticas proporciona probabilidades de que las posiciones en el bloque bidimensional de coeficientes de transformación será distinto de cero, y explorar adaptativamente el bloque bidimensional de coeficientes de transformación basados en el conjunto seleccionado de estadísticas.

30

35

El codificador de vídeo 20 también representa un ejemplo de un codificador de vídeo configurado para explorar un bloque bidimensional de coeficientes de transformación para producir un vector unidimensional de los coeficientes de transformación, determinar los valores indicativos de si los coeficientes de transformación en el vector unidimensional son significativos, y codificar por entropía al menos uno de los valores utilizando un modelo de contexto seleccionado sobre la base de al menos un porcentaje de los coeficientes significativos en un número predeterminado de los valores codificados antes de del al menos uno de los valores.

40

En algunos ejemplos, la unidad de transformación 52 puede estar configurada para ajustar a cero ciertos coeficientes de transformación (es decir, coeficientes de transformación en ciertas posiciones). Por ejemplo, la unidad de transformación 52 puede estar configurada para ajustar a cero todos los coeficientes de transformación fuera del cuadrante superior izquierdo de la TU después de la transformación. Como otro ejemplo, la unidad de codificación de entropía 56 puede configurar a cero los coeficientes de transformación de la matriz después de una cierta posición en el matriz. En algunos ejemplos, la unidad de codificación de entropía 56 puede cuantificar una matriz de dos dimensiones, y la unidad de codificación de entropía 56 puede realizar la exploración. En cualquier caso, el codificador de vídeo 20 puede estar configurado para ajustar a cero una cierta porción de los coeficientes de transformación, por ejemplo, antes o después de la exploración. La frase "a cero" se usa para significar el ajuste del valor del coeficiente igual a cero, pero no necesariamente saltar o descartar el coeficiente.

45

50

Después de la cuantificación, la unidad de codificación de entropía 56 codifica por entropía los coeficientes de transformación cuantificados. Por ejemplo, la unidad de codificación de entropía 56 puede realizar el contenido de codificación adaptativa de longitud variable (CAVLC), la codificación aritmética binaria de contexto adaptativo (CABAC), u otra técnica de codificación de entropía. Después de la codificación por entropía mediante la unidad de codificación de entropía 56, el vídeo codificado se puede transmitir a otro dispositivo o archivar para su posterior transmisión o recuperación. En el caso de codificación aritmética binaria de contexto adaptativo, el contexto puede estar basado en bloques vecinos.

55

En algunos casos, la unidad de codificación de entropía 56 u otra unidad del codificador de vídeo 20 pueden estar configuradas para realizar otras funciones de codificación, además de la codificación de entropía. Por ejemplo, la unidad de codificación de entropía 56 puede estar configurada para determinar los valores de los patrones de bloque codificados (CBP) para los bloques. Además, en algunos casos, la unidad de codificación de entropía 56 puede llevar a cabo la codificación de larga extensión de los coeficientes en un bloque.

La unidad de cuantificación inversa 58 y unidad de transformación inversa 60 aplican cuantificación inversa y transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque de predicción de uno de los fotogramas de la memoria de fotogramas de referencia 64. La unidad de compensación de movimiento 44 también puede solicitar uno o más filtros de interpolación al bloque residual reconstruido para calcular sub-valores de píxeles enteros para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción de movimiento compensado producido por la unidad de compensación de movimiento 44 para producir un bloque de vídeo reconstruido para su almacenamiento en la memoria de fotogramas de referencia 64. El bloque de vídeo reconstruido puede ser utilizado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 como un bloque de referencia para inter-codificar un bloque en un fotograma de vídeo posterior.

La figura 3 es un diagrama de bloques que ilustra un ejemplo de decodificador de vídeo 30, que decodifica una secuencia de vídeo codificada. En el ejemplo de la figura 3, el decodificador de vídeo 30 incluye una unidad de decodificación de entropía 70, una unidad de compensación de movimiento 72, la unidad de intra-predicción 74, una unidad de cuantificación inversa 76, una unidad de transformación inversa 78, una memoria de fotogramas de referencia 82 y un sumador 80. El decodificador de vídeo 30 puede, en algunos ejemplos, realizar un pase de decodificación generalmente recíproco al pase de codificación descrito respecto al codificador de vídeo 20 (figura 2). La unidad de compensación de movimiento 72 puede generar datos de predicción basados en vectores de movimiento recibidos de la unidad de decodificación de entropía 70. La unidad de intra-predicción 74 puede generar datos de predicción para un bloque actual de un fotograma actual basado en un modo de intra-predicción señalado y los datos de los bloques previamente decodificados del fotograma actual.

En algunos ejemplos, la unidad de decodificación de entropía 70 o la unidad de cuantificación inversa 76 pueden explorar los valores recibidos utilizando un reflejo de exploración utilizado por el codificador de vídeo 20. En el ejemplo de la figura 3, el decodificador de vídeo 30 incluye datos de mapeo 84, que pueden incluir datos similares o idénticos a los datos de mapeo 66. Por consiguiente, el decodificador de vídeo 30 puede seleccionar un análisis basado en, por ejemplo, una indicación de un modo de intra-codificación para un bloque actual (por ejemplo, presentado en el nodo raíz de un árbol cuádruple para la LCU que incluye el bloque actual), una transformación para el bloque actual, una transformación en cascada para el bloque actual, u otros factores utilizados por el codificador de vídeo 20 para seleccionar la exploración. Asimismo, el decodificador de vídeo 30 puede estar configurado para realizar una exploración adaptativa, o para seleccionar una exploración predeterminada basada en estos factores. De esta manera, el decodificador de vídeo 30 puede producir una matriz bidimensional de coeficientes de transformación cuantificados a partir de una matriz unidimensional de coeficientes recibidos.

La unidad de cuantificación inversa 76 cuantifica a la inversa, es decir, decuantifica, los coeficientes de transformación cuantificados proporcionados en el flujo de bits y decodificados mediante la unidad de decodificación de entropía 70. El proceso de cuantificación inversa puede incluir un proceso convencional, por ejemplo, tal como se define por el estándar de decodificación H. 264 o por HEVC. El proceso de cuantificación inversa puede incluir el uso de una cuantificación, un parámetro QPY calculado por el codificador de vídeo 20 para la CU para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que se debe aplicar.

La unidad de transformación inversa 58 aplica una transformación inversa, por ejemplo, una DCT inversa, unas transformaciones de número entero inversas, una transformación inversa de rotación o una transformación inversa direccional. En algunos ejemplos, la unidad de transformación inversa 78 puede determinar una transformación inversa basada en un modo de intra-predicción señalado para un bloque codificado con intra-predicción recibido. Si el bloque es de un tamaño para el que más de una transformación es posible, basándose en el modo de intra-predicción, entonces la unidad de transformación inversa 78 puede determinar una transformación para aplicar al bloque actual sobre la base de una transformación señalada en el nodo raíz de un árbol cuádruple para una LCU que incluye el bloque actual. En algunos ejemplos, la unidad de transformación inversa 78 puede aplicar una transformación inversa en cascada, por ejemplo, primero una transformación de rotación inversa seguida por una transformación inversa direccional.

En algunos ejemplos, por ejemplo, cuando el modo de intra-predicción señalado es el modo de predicción DC, la unidad de transformación inversa 58 (u otra unidad del decodificador de vídeo 30) puede determinar si un borde está presente en el bloque actual. La unidad de transformación inversa 58 puede determinar si el borde está presente utilizando técnicas que corresponden sustancialmente a las descritas respecto al codificador de vídeo 20 de la figura 2. Si un borde está presente en el bloque actual, la unidad de transformación inversa 78 puede determinar un ángulo del borde dentro del bloque y seleccionar una transformación inversa que se mapea a un modo de intra-predicción que tiene un ángulo que se aproxima al ángulo del borde.

Como se ha expuesto anteriormente, los datos de mapeo 84 pueden proporcionar ángulos para los modos de intra-predicción y mapeos entre los modos de intra-predicción y transformaciones inversas. En consecuencia, la unidad de transformación inversa 78 puede consultar los datos de mapeo 84 para determinar una transformación inversa mapeada a un modo de intra-predicción que tiene un ángulo que se aproxima mucho al ángulo del borde, cuando se
 5 señaliza el modo de predicción DC. Además, la unidad de intra-predicción 74 puede aplicar un modo de predicción basado en bordes para predecir el bloque, en lugar del modo de predicción DC como una señal para el bloque cuando se detecta el borde en el bloque. Los datos de mapeo 84 también pueden proporcionar un mapeo desde un modo de intra-predicción, una transformación secundaria tal como una transformación de rotación, o una combinación de los mismos, a un índice de exploración, para seleccionar una exploración inversa para recibir
 10 coeficientes de transformación cuantificados.

La unidad de compensación de movimiento 72 produce bloques de movimiento compensado, posiblemente, realizando una interpolación basada en filtros de interpolación. Los identificadores para los filtros de interpolación que se utilizan para la estimación de movimiento con una precisión de sub-píxel se pueden incluir en los elementos de sintaxis. La unidad de compensación de movimiento 72 puede utilizar filtros de interpolación y usarlos por el
 15 codificador de vídeo 20 durante la codificación del bloque de vídeo para calcular los valores interpolados para píxeles sub-enteros de un bloque de referencia. La unidad de compensación de movimiento 72 puede determinar los filtros de interpolación utilizados por el codificador de vídeo 20 de acuerdo a la información de sintaxis recibida y utilizar los filtros de interpolación para producir los bloques de predicción.

La unidad de compensación de movimiento 72 y la unidad de intra-predicción 74 utilizan parte de la información de sintaxis (por ejemplo, proporcionada por un árbol cuádruple) para determinar los tamaños de las LCUs utilizadas para codificar fotograma(s) de la secuencia de vídeo codificada, dividir la información que describe cómo se divide cada CU de un fotograma de la secuencia de vídeo codificado (y del mismo modo, cómo se dividen las sub-CUs), modos que indica cómo se codifica cada división (por ejemplo, intra- o inter-predicción, y para la intra-predicción de un modo de codificación de intra-predicción), uno o más fotogramas de referencia (y/o listas de referencia que
 20 contienen identificadores de los fotogramas de referencia) para cada PU inter-codificada, y otra información para decodificar la secuencia de vídeo codificado.

El sumador 80 combina los bloques residuales con los bloques de predicción correspondientes generados por la unidad de compensación de movimiento 72 o la unidad de intra-predicción 74 para formar bloques decodificados. Si se desea, un filtro de desbloqueo también puede aplicarse para filtrar los bloques decodificados para eliminar
 30 artefactos de formación de bloques. Los bloques de vídeo decodificados se almacenan en la memoria de fotogramas de referencia 82, que ofrece bloques de referencia para la compensación de movimiento posterior y también produce vídeo decodificado para su presentación en un dispositivo de visualización (como el dispositivo de visualización 32 de la figura 1).

De esta manera, el decodificador de vídeo 30 es un ejemplo de un decodificador de vídeo configurado para determinar un modo de intra-predicción a utilizar para predecir un bloque de datos de vídeo, y para la transformación inversa de datos residuales del transformador del bloque utilizando una transformación inversa mapeada en el modo de intra-predicción. El decodificador de vídeo 30 es también un ejemplo de un decodificador de vídeo configurado para recibir una indicación de un primer modo de intra-predicción en un primer conjunto de modos de intra-predicción para un bloque de datos de vídeo codificados, determinar un segundo modo de intra-predicción a partir de un conjunto de modos de intra-predicción más pequeño al que se mapea el primer modo de intra-predicción, determinar una transformación inversa direccional a la que se mapea el segundo modo de intra-predicción, y aplicar la transformación inversa direccional a las transformaciones de datos residuales del bloque.

El decodificador de vídeo 30 es también un ejemplo de un decodificador de vídeo configurado para recibir una primera indicación de un modo de intra-predicción a utilizar para decodificar un bloque de datos de vídeo, determinar si el bloque incluye un sub-bloque de un tamaño para el cual son posibles varias transformaciones en función del tamaño del sub-bloque y el modo de intra-predicción indicado, cuando el bloque incluye el sub-bloque del tamaño para el que varias transformaciones inversas son posibles en función del tamaño del sub-bloque y el modo de intra-predicción indicado, recibir una segunda indicación de una de las múltiples posibles transformaciones inversas, y transformación inversa del sub-bloque usando el indicado de las múltiples posibles transformaciones inversas.

El decodificador de vídeo 30 es también un ejemplo de un decodificador de vídeo configurado para recibir los valores indicativos de si los coeficientes de transformación codificados en un vector unidimensional recibido de los coeficientes de transformación codificados son significativos, decodificar con entropía al menos uno de los valores utilizando un modelo de contexto seleccionado basándose en al menos un porcentaje de los coeficientes significativos en un número predeterminado de los valores decodificados antes de que el al menos uno de los valores, y la exploración inversa el vector unidimensional produzcan un bloque de dos dimensiones de los
 50 coeficientes de transformación.

El decodificador de vídeo 30 es también un ejemplo de un decodificador de vídeo configurado para recibir una indicación de que los datos residuales para un bloque de datos de vídeo se transforman utilizando una primera transformación y una transformación de rotación para producir un bloque bidimensional de coeficientes de transformación, seleccionar un conjunto de estadísticas asociadas con al menos uno de la primera transformación y
 60 transformación, seleccionar un conjunto de estadísticas asociadas con al menos uno de la primera transformación y

las transformaciones de rotación, en el que el conjunto de estadísticas proporciona probabilidades de que las ubicaciones en el bloque bidimensional de coeficientes de transformación será distinto de cero, y explorando de manera adaptativa inversa uno vector dimensional recibido que incluye una versión codificada de los datos residuales para el bloque basado en el conjunto seleccionado de estadísticas para producir una matriz bidimensional de coeficientes de transformación para el bloque.

El decodificador de vídeo 30 es, además, un ejemplo de un decodificador de vídeo configurado para recibir información que indica que un modo de intra-predicción para un bloque de datos de vídeo es un modo de intra-predicción DC, determinar un ángulo de un borde en el bloque de vídeo de datos en base a la indicación de los modos de intra-predicción DC para el bloque, utilizando el bloque de transformación inversa una transformación inversa de dirección mapeada a un modo de intra-predicción direccional que tiene un ángulo que se aproxima al ángulo del borde, y decodificar el bloque de transformación inversa.

La figura 4 es un diagrama conceptual que ilustra el gráfico 104 que representa un ejemplo del conjunto de direcciones asociadas con los modos de intra-predicción. En el ejemplo de la figura 4, el bloque 106 se puede predecir a partir de los píxeles vecinos 100A-100AG (píxeles vecinos 100) en función de un modo de intra-predicción seleccionado. Las flechas 102A-102AG (flechas 102) son representativas de direcciones o ángulos asociados con varios modos de intra-predicción. El ejemplo de la figura 4 es representativo de los modos de intra-predicción proporcionados por el HM. Sin embargo, en otros ejemplos, se pueden proporcionar más o menos modos de intra-predicción. Aunque el ejemplo del bloque 106 es un bloque de 8 x 8 píxeles, en general, un bloque puede tener cualquier número de píxeles, por ejemplo, 4 x 4, 8 x 8, 16 x 16, 32 x 32, 64 x 64, 128 x 128, etc. Aunque el HM prevé una UP cuadrada, las técnicas de esta divulgación también pueden aplicarse a otros tamaños de bloque, por ejemplo, bloques N x M, donde N no es necesariamente igual a M.

Un modo de intra-predicción puede definirse de acuerdo con un ángulo de la dirección de predicción respecto a, por ejemplo, un eje horizontal que es perpendicular a los lados verticales del bloque 106. Así, cada una de las flechas 102 puede representar un ángulo particular a una dirección predicción de un modo de intra-predicción correspondiente. En algunos ejemplos, un modo de dirección de intra-predicción puede definirse por un par de enteros (dx, dy), que pueden representar la dirección del modo de intra-predicción correspondiente utilizado para la extrapolación de píxeles de contexto. Es decir, el ángulo del modo de intra-predicción puede calcularse como dy/dx. En otras palabras, el ángulo puede representarse de acuerdo con el desplazamiento horizontal dx y el desplazamiento vertical dy. El valor de un píxel en la posición (x, y) en el bloque 106 puede determinarse a partir de uno de los píxeles vecinos 100 a través del cual pasa una línea que pasa también a través de la ubicación (x, y) con un ángulo de dy/dx.

En algunos ejemplos, cada uno de los modos de intra-predicción correspondientes a ángulos representados por las flechas 102 puede mapearse a una transformación particular. Dos o más modos de intra-predicción se pueden mapear a la misma transformación, en algunos ejemplos. Las transformaciones pueden corresponder a transformaciones direccionales, KLTS, transformaciones de rotación, transformaciones de coseno discreto, transformaciones sinusoidales discretas, transformaciones de Fourier, u otras transformaciones que se seleccionan específicamente para determinados modos de intra-predicción. Tales transformaciones pueden llamarse generalmente como transformaciones "diseñadas", en que se seleccionan las transformaciones para determinados modos de intra-predicción.

Como se describió anteriormente, en algunos ejemplos, un conjunto de modos de intra-predicción puede mapearse, mediante una correspondencia de muchos a uno, a un conjunto más pequeño, por ejemplo, un subconjunto, de los modos de intra-predicción. Dicho de otra manera, los ángulos para los modos de intra-predicción en un gran conjunto de modos pueden correlacionarse con los ángulos de los modos de intra-predicción en un conjunto más pequeño de modos. En algunos ejemplos, el mapeo puede realizarse usando una fórmula matemática. Por ejemplo, la fórmula puede proporcionar un mapeo que minimiza la diferencia de ángulo de predicción absoluta entre el ángulo real de dirección de predicción, que se hace referencia aquí como α , y los ángulos de dirección de predicción de un conjunto más pequeño de los modos de intra-predicción, que se hace referencia aquí como β_i . Dada una dirección de predicción del ángulo α , la fórmula puede proporcionar un β_i de tal manera que se satisface la fórmula (1) anterior. La fórmula (1) se actualiza a continuación por conveniencia:

$$\arg\{\beta_i\} \min(\min(\text{abs}(\alpha - \beta_i), \text{abs}(-\alpha - \beta_i))) \quad (1)$$

En un ejemplo, el conjunto de modos de intra-predicción más pequeño puede tener ángulos que son los mismos que las flechas 102E, 102I, 102M, 102Q, 102U, 102Y, 102AC, y 102AG. Por lo tanto, cada uno de los ángulos de las flechas 102 puede correlacionarse con uno de los ángulos de las flechas 102E, 102I, 102M, 102Q, 102U, 102Y, 102AC, y 102AG. Como un ejemplo, los ángulos de las flechas 102A-102E pueden correlacionarse con el ángulo de la flecha 102E, los ángulos de las flechas 102F-102I pueden correlacionarse con el ángulo de la flecha 102I, los ángulos de las flechas 102J-102M pueden correlacionarse con el ángulo de la flecha 102M, los ángulos de las flechas 102N-102Q pueden correlacionarse con el ángulo de la flecha 102Q, los ángulos de las flechas 102R-102U pueden correlacionarse con el ángulo de la flecha 102U, los ángulos de las flechas 102V-102Y pueden mapearse al ángulo de la flecha 102Y, los ángulos de las flechas 102Z-102AC pueden correlacionarse con el ángulo de la flecha

102AC, y los ángulos de las flechas 102AD-102AG pueden correlacionarse con el ángulo de la flecha 102AG.

Otros mapeos también se pueden proporcionar. En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse con una variedad de diferentes mapeos, y el codificador de vídeo 20 puede proporcionar información indicativa del mapeo utilizado para un flujo de bits particular, por ejemplo, en los datos del encabezado, un conjunto de parámetros de secuencia (SPS), u otros datos señalizados.

Como se describió anteriormente, en algunos ejemplos, un dispositivo de codificación de vídeo (como el codificador de vídeo 20 o el decodificador de vídeo 30) puede estar configurado para determinar si un borde está presente en un bloque. Por ejemplo, el dispositivo de codificación de vídeo puede estar configurado para determinar si un borde está presente en el bloque 106 en base a un análisis de píxeles de uno o más bloques vecinos, donde los bloques vecinos pueden incluir uno o más de los píxeles vecinos 100. En general, un bloque vecino previamente codificado puede compartir una frontera con el bloque 106, donde la frontera puede corresponder a uno o más de los píxeles vecinos 100. Por ejemplo, un bloque vecino a la izquierda del bloque 106 puede incluir píxeles vecinos 100I-100P, que definen una frontera entre el bloque vecino de la izquierda y el bloque 106.

El dispositivo de codificación de vídeo puede configurarse para calcular los gradientes de los píxeles en un bloque vecino previamente codificado para determinar si un borde está presente en el bloque vecino previamente codificado. El dispositivo de codificación de vídeo puede determinar, además, si el borde cruza (es decir, atraviesa) una frontera entre los bloques vecinos previamente codificados y un bloque actual, como el bloque 106. Respecto al ejemplo del bloque vecino de la izquierda al bloque 106 descrito anteriormente, el dispositivo de codificación de vídeo puede determinar si los gradientes de los píxeles en el bloque vecino de la izquierda indican la presencia de un borde que cruza la frontera entre el bloque vecino de la izquierda y el bloque 106, donde la frontera está definida por los píxeles 100I-100P, en este ejemplo. Cuando el dispositivo de codificación de vídeo determina que los gradientes de los píxeles en el bloque vecino de la izquierda indican la presencia de un borde y que el borde cruza la frontera definida por los píxeles 100I-100P, el dispositivo de codificación de vídeo puede determinar que el borde procede en el bloque 106, y como tal, que el bloque 106 incluye un borde.

En algunos ejemplos, cuando el dispositivo de codificación de vídeo determina que un modo de intra-predicción señalado para el bloque 106 es el modo de predicción DC, y que el bloque 106 incluye un borde, el dispositivo de codificación de vídeo puede predecir el bloque 106 utilizando un modo de intra-predicción basado en bordes. Además, el dispositivo de codificación de vídeo puede determinar un ángulo del borde. El dispositivo de codificación de vídeo entonces puede determinar un ángulo de un modo de predicción, indicado en general por las flechas 102, que más se aproxima al ángulo del borde. El dispositivo de codificación de vídeo entonces puede seleccionar un archivo de transformación (que puede corresponder a una transformación inversa cuando se decodifica) que está mapeado al modo de intra-predicción que tiene el ángulo que más se aproxima al ángulo del borde, y aplicar el seleccionado para transformar los datos residuales del bloque 106. Durante un proceso de codificación, el dispositivo de codificación de vídeo puede aplicar la transformación a una TU del bloque 106, mientras que durante un proceso de decodificación, el dispositivo de codificación de vídeo puede aplicar una transformación inversa de datos residuales transformados para el bloque 106.

La figura 5 es un diagrama conceptual que ilustra los modos de intra-predicción 110A-110I (modos de intra-predicción 110) de H.264. El modo de intra-predicción 110C corresponde a un modo de intra-predicción de DC, y por lo tanto, no necesariamente está asociado a un ángulo real. Los modos de intra-predicción restantes 110 pueden estar asociados con un ángulo, similar a los ángulos de las flechas 102 de la figura 4. Por ejemplo, el ángulo de modo de intra-predicción 110A corresponde a la flecha 102Y, el ángulo de modo de intra-predicción 110B corresponde a la flecha 102I, el ángulo de modo de intra-predicción 110D corresponde a la flecha 102AG, el ángulo de modo de intra-predicción 110E corresponde a la flecha 102Q, el ángulo de modo de intra-predicción 110F corresponde a la flecha 102U, el ángulo de modo de intra-predicción 110G corresponde a la flecha 102M, el ángulo de modo de intra-predicción 110H corresponde a la flecha 102AD, y el ángulo de modo de intra-predicción 110I corresponde a la flecha 102E.

Los ángulos de las flechas 102 que no corresponden directamente a uno de los modos de intra-predicción 110 pueden asignarse a uno de los modos de intra-predicción 110. Por ejemplo, el ángulo de uno de los modos de intra-predicción 110 que se aproxima al ángulo de una de las flechas 102 puede corresponder al ángulo con el que una de las flechas 102 es mapeada.

Cada uno de los modos de intra-predicción 110 puede ser mapeado a una transformación particular, por ejemplo, con una correspondencia de uno a uno. Por ejemplo, un dispositivo de codificación de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, puede incluir datos de configuración que mapea el modo de intra-predicción 110C a una DCT, y cada uno de los otros modos de intra-predicción 110 a una transformación direccional particular, por ejemplo, una KLT. En consecuencia, los ángulos para cada uno de los modos de intra-predicción asociados con las flechas 102 (figura 4) pueden correlacionarse con ángulos de modos de intra-predicción 110 (figura 5). Los modos de intra-predicción 110, a su vez, pueden mapearse a transformaciones, por ejemplo, transformaciones direccionales. De esta manera, los ángulos para cada uno de los modos de intra-predicción asociados con las flechas 102 (figura 4) pueden correlacionarse con las transformaciones direccionales. En consecuencia, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar una transformación

direccional para aplicarse a una TU basada en un modo de intra-predicción seleccionado para una PU correspondiente a la TU.

La figura 6 es un diagrama conceptual que ilustra un ejemplo en exploración en zigzag de los coeficientes 120A-120P (coeficientes 20). Los coeficientes 120 generalmente corresponden a coeficientes de transformación cuantificados, resultantes de la transformación y la cuantificación de píxeles o una TU. El codificador de vídeo 20 puede estar configurado para explorar un bloque de coeficientes utilizando la exploración en zigzag de la figura 6 siguiente, por ejemplo, aplicando una DCT de un bloque residual. En este ejemplo, la exploración en zigzag comienza en el coeficiente 120A, a continuación sigue al coeficiente 120B, a continuación al coeficiente 120E, a continuación, al coeficiente 120I, a continuación, al coeficiente 120F, a continuación al coeficiente 120C, a continuación, al coeficiente 120D, a continuación, al coeficiente 120G, a continuación, al coeficiente 120J, a continuación, al coeficiente 120M, a continuación, al coeficiente 120N, a continuación, al coeficiente 120K, a continuación, a coeficiente 120H, a continuación, al coeficiente 120L, a continuación, al coeficiente 120O, y finalmente al coeficiente 120P.

Mediante la realización de este análisis, la disposición bidimensional de coeficientes para los píxeles se puede convertir en una matriz unidimensional que incluye los valores para cada uno de los coeficientes 120. Estos valores pueden estar dispuestos en la matriz en el orden de exploración. Por ejemplo, el valor para el coeficiente 120A puede ser primero en la matriz, seguido por valores de coeficientes 120B, 120E, 120I, 120F, y así sucesivamente.

Otros patrones de exploración predefinidos también pueden ser definidos mediante otras transformaciones. Por ejemplo, cada transformación direccional puede estar asociada con un patrón de exploración que está diseñado para colocar coeficientes de baja frecuencia resultantes de la anterior transformación direccional en una matriz de coeficientes de frecuencia más altos. Una de las transformaciones direccionales puede hacer que se produzcan coeficientes de frecuencias más bajas a lo largo de la columna del extremo izquierdo de un bloque de coeficientes de transformación, en cuyo caso se puede definir una exploración correspondiente que se inicia en el coeficiente 120A, a continuación, sigue al coeficiente 120E, a continuación, al coeficiente 1,201, a continuación, al coeficiente 120M, a continuación, a 120B, y así sucesivamente. Como otro ejemplo, otra de las transformaciones direccionales puede hacer que se produzcan coeficientes de frecuencias más bajas a lo largo de la fila superior de un bloque de coeficientes de transformación, en cuyo caso se puede definir una exploración correspondiente que se inicia en el coeficiente 120A, a continuación, se procede al coeficiente 120B, a continuación al coeficiente 120C, a continuación, al coeficiente 120D, a continuación, al coeficiente 120E, y así sucesivamente.

En algunos ejemplos, el codificador de vídeo 20 puede estar configurado para realizar una exploración adaptativa, en lugar de una exploración predefinida. La exploración adaptativa puede variar con el tiempo en base a las estadísticas indicativas de si los coeficientes particulares (es decir, los coeficientes correspondientes a los coeficientes 120) son significativos. Por otra parte, el codificador de vídeo 20 podrá calcular conjuntos separados de estadísticas basadas en, por ejemplo, un modo de intra-predicción seleccionado para predecir un bloque, un índice de una transformación de rotación para aplicarse después de una transformación inicial, u otros factores.

En algunos ejemplos, el codificador de vídeo 20 puede incluir dos tablas para estas estadísticas: una tabla de alto rango dinámico y una tabla de exploración de actualización dinámica. Suponiendo que el bloque que se va a explorar tiene coeficientes $N \times N$, cada una de estas dos tablas también pueden tener tamaños de $N \times N$. La tabla de alto rango dinámico puede ser una tabla predeterminada fija que proporciona probabilidades de que cada coeficiente de transformación es diferente de cero. Esta tabla se puede calcular sobre la base de un conjunto de datos de entrenamiento. Por otra parte, esta tabla puede ser utilizada para proporcionar el punto de partida para la orden de exploración adaptativa. En general, la tabla de alto rango dinámico puede ser estática (es decir, sin cambios) para un flujo de bits.

La tabla de exploración de actualización dinámica se puede actualizar con el tiempo para reflejar determinadas estadísticas de coeficientes de transformación recientes. En particular, el codificador de vídeo 20 puede mantener un recuento del número de veces que cada coeficiente es distinto de cero. Es decir, para cada bloque de transformación, el codificador de vídeo 20 puede incrementar los valores en la tabla de exploración de actualización dinámica correspondiente a los coeficientes diferentes de cero en el bloque de transformación actual. Por ejemplo, si el coeficiente de transformación correspondiente al coeficiente 120E no es cero, el codificador de vídeo 20 podrá añadir uno al valor en la tabla de exploración de actualización dinámica correspondiente al coeficiente 120E. Los valores en la tabla de exploración de actualización dinámica también pueden ser normalizados periódicamente, para evitar que los valores excedan de un valor máximo.

Para realizar la exploración de adaptación para una primera unidad de transformación de un fotograma, el codificador de vídeo 20 puede basarse en el barrido en la tabla de alto rango dinámico solamente. El codificador de vídeo 20 puede inicializar la tabla de exploración de actualización dinámica mediante, por ejemplo, el establecimiento de todos los valores en la tabla de exploración de actualización dinámica a cero. Para cada coeficiente distinto de cero en el bloque de transformación, el codificador de vídeo 20 también puede añadir uno a los valores yuxtapuestas en la tabla de exploración de actualización dinámica. Para los bloques posteriores, el codificador de vídeo 20 puede referirse primero a la tabla de exploración de actualización dinámica para determinar cuál de los coeficientes de transformación es más probable que sea distinto de cero, a continuación, se analiza en

orden decreciente de probabilidad de coeficientes que son distintos de cero. En algunos casos, dos o más valores de la tabla de exploración de actualización dinámica pueden ser iguales. En tal caso, unidad de cuantificación 54 se refiere a la tabla de alto rango dinámico para determinar cuál de los coeficientes se explorará a continuación. De esta manera, la unidad de cuantificación 54 puede realizar un análisis adaptativo basado en una combinación de una

La tabla de alto rango dinámico puede ser la misma para todas las estadísticas de exploración adaptativa, en algunos ejemplos. Por lo tanto, el codificador de vídeo 20 puede incluir la actualización dinámica de las tablas de exploración específicas para, por ejemplo, el modo de infra-predicción seleccionado, el índice de transformación de rotación, o una combinación de los mismos. En algunos ejemplos, el codificador de vídeo 20 puede estar configurado para seleccionar entre exploraciones estáticas predeterminadas cuando una transformación de rotación no se aplica, y para realizar una exploración adaptativa cuando se aplica una transformación de rotación y, además, para seleccionar las estadísticas para realizar la exploración adaptativa basada en uno o ambos del modo de intra-predicción seleccionado y el índice de la transformación de rotación seleccionada. Alternativamente, el codificador de vídeo 20 puede estar configurado para seleccionar una exploración predefinida sobre la base de un índice de la transformación de rotación, un modo de intra-predicción, o una combinación de los mismos. El decodificador de vídeo 30 puede estar configurado de manera similar al codificador de vídeo 20 para seleccionar una exploración adecuada.

Las figuras 7A y 7B son diagramas conceptuales que ilustran un ejemplo de árbol cuádruple 150 y una unidad de codificación 172 más grande correspondiente. La figura 7A representa un ejemplo de árbol cuádruple 150, que incluye nodos dispuestos de forma jerárquica. Cada nodo en un árbol cuádruple, como el árbol cuádruple 150, puede ser un nodo hoja sin secundarios, o tiene cuatro nodos secundarios. En el ejemplo de la figura 7A, el árbol cuádruple 150 incluye nodo raíz 152. El nodo raíz 152 tiene cuatro nodos secundarios, incluyendo nodos de hoja 156A-156C (nodos de hoja 156) y el nodo 154. Como el nodo 154 no es un nodo hoja, el nodo 154 incluye cuatro nodos secundarios, que en este ejemplo, son nodos de hoja 158A-158D (nodos de hoja 158).

El árbol cuádruple 150 puede incluir datos que describen características de una unidad de codificación más grande correspondiente (LCU), tales como LCU 172 en este ejemplo. Por ejemplo, el árbol cuádruple 150, por su estructura, pueden describir la división de la LCU en sub-CUs. Supongamos que LCU 172 tiene un tamaño de $2N \times 2N$. La LCU 172, en este ejemplo, tiene cuatro sub-CUs 176A-176C (sub-CU 176) y 174, cada uno de tamaño $N \times N$. La sub-CU 174 está aún más dividida en cuatro sub-CUs 178A-178D (sub-CU 178), cada una de tamaño $N/2 \times N/2$. La estructura de árbol cuádruple 150 corresponde a la división de LCU 172, en este ejemplo. Es decir, el nodo raíz 152 corresponde a LCU 172, los nodos de hoja 156 corresponden a la sub-CU 176, el nodo 154 corresponde a la sub-CU 174, y los nodos de hoja 158 corresponden a la sub-CU 178.

Los datos para los nodos de árbol cuádruple 150 pueden describir si la CU correspondiente al nodo está dividida. Si la CU está dividida, cuatro nodos adicionales pueden estar presentes en árbol cuádruple 150. En algunos ejemplos, un nodo de un árbol cuádruple puede implementarse similar al siguiente pseudocódigo:

```
quadtree_node {
    boolean split_flag(1);
    // signaling data
    if(split_flag) {
        quadtree_node child1;
        quadtree_node child2;
        quadtree_node child3;
        quadtree_node child4;
    }
}
```

El valor `split_flag` puede ser un valor representativo de un bit de si la CU correspondiente al nodo actual está dividido. Si la CU no está dividida, el valor `split_flag` puede ser "0", mientras que si la CU está dividida, el valor `split_flag` puede ser "1". Respecto al ejemplo de árbol cuádruple 150, una matriz de valores de señal de división puede ser 101000000.

En algunos ejemplos, cada uno de las sub-CUs 176 y sub-CUs 178 puede ser codificada con intra-predicción utilizando el mismo modo de intra-predicción. Por consiguiente, el codificador de vídeo 20 puede proporcionar una indicación del modo de intra-predicción en el nodo raíz 152. Además, ciertos tamaños de sub-CUs pueden tener varias transformaciones posibles para un modo de intra-predicción particular. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 puede proporcionar una indicación de la transformación para usar para tales sub-CUs en el nodo raíz 152. Por ejemplo, las sub-CUs de tamaño $N/2 \times N/2$ pueden tener varias transformaciones posibles disponibles. El codificador de vídeo 20 puede señalar la transformación a utilizar en el nodo raíz 152. En consecuencia, el decodificador de vídeo 30 puede determinar la transformación a aplicar a las sub-CUs 178 basándose en el modo de intra-predicción señalado en el nodo raíz 152 y la transformación señalada en el nodo raíz 152.

Como tal, codificador de vídeo 20 no necesita señalar las transformaciones a aplicar a las sub-CUs 176 y las sub-CUs 178 en los nodos de hojas 156 y los nodos de hoja 158, pero podrá limitarse a señalar un modo de intra-predicción y, en algunos ejemplos, una transformación para aplicarse a ciertos tamaños de sub-CUs, en el nodo raíz 152, de acuerdo con las técnicas de esta divulgación. De esta manera, estas técnicas pueden reducir los gastos generales de señalización de las funciones de transformación para cada sub-CU de una LCU, tales como LCU 172.

En algunos ejemplos, los modos de intra-predicción para las sub-CUs 176 y/o las sub-CUs 178 pueden ser diferentes de los modos de intra-predicción para LCU 172. El codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar con las funciones que se mapean en el modo de intra-predicción marcado en el nodo raíz 152 a un modo de intra-predicción disponible para las sub-CUs 176 y/o sub-CUs 178. La función puede proporcionar un mapeo de los modos de intra-predicción de muchos a uno disponibles para la LCU 172 a modos de intra-predicción para las sub-CUs 176 y/o las sub-CUs 178.

Aunque la figura 7A ilustra un ejemplo de un árbol cuádruple de CU, debe entenderse que un árbol cuádruple similar puede aplicarse a unidades de formación de una CU de nodo hoja. Es decir, una CU de nodo hoja puede incluir un árbol cuádruple de TU que describe la partición de unidades de formación para la CU. El árbol cuádruple de ATU generalmente puede parecerse a un árbol cuádruple de CU, excepto que el árbol cuádruple de TU puede indicar modos de intra-predicción para unidades de formación de la CU individualmente.

La figura 8 es un diagrama de flujo que ilustra un procedimiento de ejemplo para la selección de una transformación y una exploración para aplicar a un bloque basado en un modo de intra-predicción seleccionado para el bloque. Aunque se ha descrito generalmente como realizado por componentes del codificador de vídeo 20 (figura 2) para fines de explicación, debe entenderse que otras unidades de codificación de vídeo, tales como procesadores, unidades de procesamiento basadas en hardware, tales como unidades de codificadores/decodificadores (CODECs), y similares, también se puede configurar para realizar el procedimiento de la figura 8.

En el ejemplo de la figura 8, la unidad de transformación 52 puede inicialmente recibir datos residuales de una TU actual (180). Además, la unidad de transformación 52 también puede recibir una indicación de un modo de intra-predicción seleccionado para la TU. A partir de esta indicación, la unidad de transformación 52 puede determinar una dirección de predicción de la TU (182). Por ejemplo, la unidad de transformación 52 puede determinar un ángulo de la dirección de predicción para el modo de intra-predicción indicado.

En cualquier caso, después de determinar el modo de intra-predicción, la unidad de transformación 52 puede seleccionar una transformación para aplicarse a los datos residuales en base a un mapeo desde el modo de intra-predicción para la transformación (186). Por ejemplo, la unidad de transformación 52 puede seleccionar la transformación para aplicarse mediante los datos de mapeo de consulta 66 con la dirección de intra-predicción y determinar la transformación a la que se mapea la dirección dentro de la predicción. La transformación puede corresponder a una transformación discreta del coseno o una transformación direccional, como una transformación direccional dependiente del modo (MDDT). La unidad de transformación 52 puede entonces aplicar la transformación seleccionada a los datos residuales para transformar los datos residuales (188). En algunos ejemplos, los datos de mapeo 66 pueden incluir además una indicación de a qué unidad de transformación 52 debe aplicarse dos o más transformaciones, tal como una transformación de rotación después de la primera transformación, en cuyo caso la unidad de transformación 52 puede aplicar también la transformación de rotación indicada.

Mediante la transformación de los datos residuales, la unidad de transformación 52 puede producir una matriz bidimensional de coeficientes de transformación que tienen el mismo número de coeficientes que los datos residuales. En el ejemplo de la figura 8, la unidad de cuantificación 54 puede entonces cuantificar los coeficientes de transformación (190). En algunos ejemplos, la unidad de cuantificación 54 puede explorar la matriz bidimensional de coeficientes para producir una matriz unidimensional, por ejemplo, antes o después de la cuantificación de los coeficientes. Alternativamente, la unidad de codificación de entropía 56 puede explorar la matriz de dos dimensiones.

En este ejemplo, la unidad de codificación de entropía 56 puede consultar los datos de mapeo 66 para seleccionar una exploración para aplicarse a los coeficientes de transformación cuantificados (192). En algunos ejemplos, los datos de mapeo 66 pueden incluir datos de mapeo de modos de intra-predicción a patrones de análisis particulares predefinidos. En algunos ejemplos, los datos de mapeo 66 pueden incluir datos que mapean transformaciones a los patrones de análisis predefinidos. En algunos ejemplos, por ejemplo, donde los datos de mapeo 66 indican que una transformación rotación es para aplicarse a coeficientes de transformación, los datos de mapeo 66 pueden indicar, además, que debe realizarse una exploración adaptativa, o se mapea una exploración predefinida a la transformación de rotación. En los ejemplos en los cuales se realiza una exploración de adaptación, los datos de mapeo 66 puede incluir también estadísticas de análisis, por ejemplo, una tabla de alto rango dinámico y una tabla de exploración de actualización dinámica, mapeada al modo de intra-predicción, un índice de las primeras transformaciones, un índice de transformación de rotación, una combinación de los mismos, y/u otros factores.

La unidad de codificación de entropía 56 puede entonces explorar los coeficientes de transformación cuantificados utilizando la exploración seleccionado (194), por ejemplo, la exploración predefinida o la exploración adaptativa

basada en las estadísticas de análisis seleccionadas. En algunos ejemplos, la unidad de codificación de entropía 56 puede estar configurada con una posición de exploración (que puede ser menor o igual al número de coeficientes de transformación) después de lo cual la unidad de codificación de entropía 56 puede ajustar a cero los valores de los coeficientes de la matriz. Después de explorar una serie de coeficientes iguales a la posición de exploración, la unidad de codificación de entropía 56 puede establecer valores de matriz restantes iguales a cero. La puesta a cero de los coeficientes de transformación puede realizarse antes o después de la exploración, en algunos ejemplos.

En algunos ejemplos, la unidad de codificación de entropía 56 puede entonces codificar por entropía los coeficientes escaneados en la matriz después de la exploración (196). Alternativamente, en algunos ejemplos, la unidad de codificación de entropía 56 puede codificar por entropía los coeficientes que se exploran. En cualquier caso, la unidad de codificación de entropía 56 puede utilizar CABAC o CAVLC para la codificación por entropía de los coeficientes.

Cuando se usa CABAC, y al realizar una exploración adaptativa, la unidad de codificación de entropía 56 puede codificar por entropía elementos de sintaxis que incluyen señales de coeficiente significativo y señales de los últimos coeficientes. La unidad de codificación de entropía 56 puede seleccionar modelos de contexto para la codificación de las señales de coeficiente de entropía significativas en base a un tipo de bloque (intra o inter), un modo de intra-predicción seleccionada (suponiendo que el bloque se prevé en un modo intra), y/o un tipo de transformación aplicada (por ejemplo, DCT o direccional/KLT). La unidad de codificación de entropía 56 puede seleccionar los modelos de contexto para la codificación por entropía de las últimas señales coeficiente en base al índice de orden en la exploración adaptativa, el tipo de bloque, la dirección de predicción espacial, y/o una transformación seleccionada.

De esta manera, el procedimiento de la figura 8 representa un ejemplo de un procedimiento que incluye el cálculo de un bloque residual para un bloque de datos de vídeo sobre la base de un bloque predicho formado usando un modo de intra-predicción, y transformar el bloque residual usando una transformación mapeada al modo de intra-predicción.

La figura 9 es un diagrama de flujo que ilustra otro ejemplo del procedimiento para la selección de una transformación y exploración para aplicarse a un bloque de datos residuales. En general, la figura 9 se ajusta sustancialmente a la figura 8. Sin embargo, en el ejemplo de la figura 9, después de recibir los datos residuales (180) e indicar un modo de intra-predicción seleccionado para la TU, la unidad de transformación 52 puede determinar una primera dirección de predicción para predecir la TU (183). Por ejemplo, la unidad de transformación 52 puede determinar un ángulo de la dirección de predicción para el modo de intra-predicción indicado.

La unidad de transformación 52 puede entonces determinar una segunda dirección mapeada a la primera dirección de predicción (184). Por ejemplo, la unidad de transformación 52 puede consultar los datos de mapeo 66 para determinar un segundo modo de intra-predicción al que se mapea el primer modo de intra-predicción. En algunos ejemplos, la unidad de transformación 52 puede determinar un ángulo que se aproxima al ángulo del modo de intra-predicción indicado, y seleccionar un segundo modo de intra-predicción correspondiente al ángulo determinado. La unidad de transformación 52 puede a continuación seleccionar una transformación mapeada a los segundos datos de predicción (185). Después de seleccionar la transformación, que puede corresponder a la selección de varias transformaciones, el codificador de vídeo 20 generalmente realiza las etapas restantes de la figura 9 de una manera similar a las etapas correspondientes descritas respecto a la figura 8.

De esta manera, el procedimiento de la figura 9 es un ejemplo de un procedimiento que incluye la recepción de una indicación de un primer modo de intra-predicción en un primer conjunto de modos de intra-predicción para un bloque de datos de vídeo, la determinación de un segundo modo de intra-predicción a partir de un conjunto más pequeño de los modos de intra-predicción para el que se mapea el primer modo de intra-predicción, la determinación de una transformación direccional al que se mapea el segundo modo de intra-predicción, y la aplicación de la transformación direccional para los datos residuales del bloque.

La figura 10 es un diagrama de flujo que ilustra un ejemplo de procedimiento para aplicar un modo de intra-predicción y transformar a sub-UCs de tamaño particular. Aunque se ha descrito generalmente como realizado mediante componentes del codificador de vídeo 20 (figura 2) para fines de explicación, debe entenderse que otras unidades de codificación de vídeo, tales como procesadores, unidades de procesamiento, unidades de codificación basadas en hardware, tales como codificadores/decodificadores (CODECs), y similares, también se puede configurar para realizar el procedimiento de la figura 10. También debe entenderse que en otros ejemplos, procedimientos similares pueden incluir etapas adicionales o alternativas a las ilustradas en la figura 10, o se pueden realizar las etapas ilustradas en un orden diferente, sin apartarse de las técnicas descritas.

Las técnicas para seleccionar y aplicar una transformación tal como se describe respecto a la figura 10 pueden corresponder a las etapas 186 y 188 de la figura 8. Las técnicas para la aplicación de diversos modos de intra-predicción a bloques de diferentes tamaños como se describe respecto a la figura 10 se pueden realizar antes de la etapa 180 de la figura 8.

La unidad de intra-predicción 46 puede recibir un bloque de píxeles, por ejemplo, una LCU (200). La unidad de intra-

predicción 46 puede determinar entonces un modo de intra-predicción para aplicarse a la LCU y señalar el modo de intra-predicción determinado para la LCU (201), por ejemplo, en un nodo raíz de una estructura de datos de árbol cuádruple correspondiente a la LCU. La unidad de intra-predicción 46 puede determinar entonces tamaños de sub-CUs para los que sólo está disponible un subconjunto de modos de intra-predicción de arco (202), la unidad de intra-predicción 46 puede además la partición de la LCU en una o más sub-CUs y determinar si alguna de las sub-CUs tiene un tamaño para los que sólo un subconjunto de los modos de intra-predicción están disponibles (204).

Si la LCU incluye sub-CUs de un tamaño para el que sólo un subconjunto de los modos de intra-predicción de arco está disponible (rama "SI" de 184), la unidad de intra-predicción 46 puede intra-predecir la sub-CU utilizando un modo de intra-predicción al que se mapea el modo de intra-predicción seleccionado para la LCU (206). Por otro lado, si la LCU no incluye ninguna sub-CU que son de tal tamaño (rama "NO" de 184), la unidad de intra-predicción 46 puede aplicar el modo señalado para la LCU a todos los sub-bloques de la LCU (208).

El codificador de vídeo 20 puede luego calcular los valores residuales para las sub-CUs de la LCU. Entonces, la unidad de transformación 32 puede determinar el tamaño de la sub-CU para los que son posibles varias transformaciones basadas en el modo de intra-predicción señalado para la LCU (210). La unidad de transformación 52 puede determinar, además, si cualquiera de las sub-CUs de la LCU son de un tamaño para el que son posibles varias transformaciones (212). Si al menos una sub-CU es de un tamaño para el que varias transformaciones son posibles (rama "SI" de 212), la unidad de transformación 52 puede seleccionar y señalar una transformación para solicitar la sub-CU de ese tamaño (214). Por ejemplo, la unidad de transformación 52 puede indicar la transformación a aplicar a las sub-CUs de ese tamaño en el nodo raíz del árbol cuádruple para la LCU. La unidad de transformación 52 también puede solicitar la transformación señalizada a todos los sub-bloques en la LCU de ese tamaño (216). Por otro lado, si la LCU no contiene ninguna sub-CU para la que son posibles varias transformaciones (rama "NO" de 212), la unidad de transformación 52 puede aplicar transformaciones a la sub-CU de la LCU basada puramente en el modo de intra-predicción señalado para la LCU (218), de tal manera que no son necesarias transformaciones respecto a la señalización.

De esta manera, el procedimiento de la figura 10 es un ejemplo de un procedimiento que incluye la selección de un modo de intra-predicción para utilizarse para codificar un bloque de datos de vídeo, determinar si el bloque incluye un sub-bloque de un tamaño para que varias transformaciones sean posibles en base al tamaño del sub-bloque y el modo de intra-predicción seleccionado, cuando el bloque incluye el sub-bloque del tamaño para el que varias transformaciones son posibles en base al tamaño del sub-bloque y el modo de intra-predicción seleccionado seleccionando una de las varias transformaciones posibles, la transformación del sub-bloque con el seleccionado de las posibles varias transformaciones, y proporcionando una indicación de la seleccionado de las varias transformaciones posibles para el tamaño del bloque.

La figura 11 es un diagrama de flujo que ilustra un procedimiento de ejemplo para realizar una exploración adaptativa basada en una transformación seleccionada aplicada a un bloque. Aunque se ha descrito generalmente como realizado por componentes del codificador de vídeo 20 (figura 2) para fines de explicación, debe entenderse que otras unidades de codificación de vídeo, tales como procesadores, unidades de procesamiento, unidades de codificación basadas en hardware, tales como codificadores/decodificadores (CODECs), y similares, también se puede configurar para realizar el procedimiento de la figura 11.

Debe entenderse también que en otros ejemplos, procedimientos similares pueden incluir etapas adicionales o alternativas a las ilustradas en la figura 11, o pueden realizar las etapas ilustradas en un orden diferente, sin apartarse a las técnicas descritas. Cuatro técnicas de coeficientes de exploración adaptativa siguiendo transformaciones en cascada como se ilustra en la figura 11 pueden corresponder a las etapas 192 y 194 de la figura 8. Las técnicas para explorar adaptativamente los coeficientes residuales de la figura 11 pueden aplicarse a los datos residuales siguiendo la intra-predicción o inter-predicción.

En un principio, la unidad de transformación 52 del codificador de vídeo 20 puede recibir un bloque residual (230). El bloque residual puede corresponder a la intra-predicción o inter-predicción residual posterior de una CU. El bloque residual puede ser del mismo tamaño o de un tamaño diferente de una unidad de predicción correspondiente de la CU. La unidad de transformación 52 puede entonces transformar el bloque residual (232). En algunos ejemplos, la unidad de transformación 52 puede aplicar una transformación direccional correspondiente a un modo de intra-predicción a partir de un subconjunto de modos de intra-predicción, de acuerdo con las técnicas de esta divulgación. En otros ejemplos, la unidad de transformación 52 puede aplicar una transformación de coseno discreta.

La unidad de transformación 52 puede entonces aplicar una transformación de rotación al bloque de transformación (234). Para unidades de transformación (TUs) de tamaños 8 x 8 y más grandes, para que unidad de transformación 52 aplique una DCT, la unidad de transformación 52 puede aplicar la transformación de rotación para los coeficientes DCT de 8 x 8 de frecuencia más baja. Para TUs menores de 8 x 8, la unidad de transformación 52 puede aplicar la transformación de rotación para toda el TU. Si la PU correspondiente a la TU fue codificada con intra-predicción, la unidad de transformación 52 puede seleccionar una transformación de rotación basándose en el modo de intra-predicción utilizado para predecir la PU, por ejemplo, cuando un subconjunto de modos de intra-predicción puede mapearse a las transformaciones de rotación. Estas transformaciones de rotación pueden denominarse como el modo dependiente de transformaciones de rotación (MDROTs). En algunos ejemplos, la

unidad de transformación 52 puede conectar en cascada una transformación de rotación (u otra transformación separable secundaria) después de un modo dependiente de transformación direccional (MDDT), que puede ser un KLT.

Después de la transformación, la unidad de cuantificación de rotación 54 puede cuantificar los coeficientes transformados, en algunos ejemplos. Entonces, la unidad de codificación de entropía 56 puede seleccionar un conjunto de estadísticas a utilizar para realizar un análisis de adaptación de los coeficientes de transformación. La lista de estadísticas puede incluir una tabla de alto rango dinámico (HDR) y una tabla de actualización dinámica (DU). Una o ambas de las tablas de HDR y DU puede seleccionarse para un escenario particular, por ejemplo, si se ha utilizado intra-predicción o inter-predicción para predecir una PU, un modo de intra-predicción en particular para la PU cuando se utiliza intra-predicción, si una DCT o KLT se aplicó a la TU correspondiente a la PU, el índice de la transformación de rotación utilizada, o cualquier combinación de los mismos de esta manera, la unidad de codificación de entropía 56 puede seleccionar las tablas de HDR y/o DU para su uso durante la exploración adaptativa (236).

Como se describió anteriormente, la tabla HDR puede incluir un conjunto predefinido de datos que indican las probabilidades de que los coeficientes de cada posición en una matriz no sean cero. La tabla HDR se puede producir usando un conjunto de datos de entrenamiento, y puede seguir siendo la misma a través de todo un flujo de bits. La unidad de codificación de entropía 56 puede recopilar estadísticas individuales a un fotograma, porción, grupo de imágenes, u otra unidad de datos de vídeo para calcular los valores de la tabla DU. La tabla DU, por lo tanto, también puede indicar las probabilidades de que los coeficientes de cada posición en la matriz no sean cero.

Para realizar la exploración adaptativa, la unidad de codificación de entropía 56 puede determinar en primer lugar la ubicación en la matriz que tiene la más alta probabilidad de incluir un coeficiente distinto de cero utilizando la tabla DU (238). En algunos casos, puede haber dos o más ubicaciones en la matriz con probabilidades iguales de tener coeficientes distintos de cero. Por lo tanto, la unidad de codificación de entropía 56 puede determinar si hay múltiples ubicaciones en la matriz con las mismas probabilidades de inclusión de coeficientes distintos de cero (240).

Si hay varias ubicaciones en la matriz con las mismas probabilidades de incluir coeficientes distintos de cero (rama "Sí" de 240), la unidad de codificación de entropía 56 puede determinar la ubicación en la matriz que tiene la más alta probabilidad de incluir un coeficiente distinto de cero usando la tabla HDR (242).

La unidad de codificación de entropía 56 puede a continuación analizar y codificar el coeficiente de entropía en la ubicación determinada (244). La unidad de codificación de entropía 56 también puede determinar si el coeficiente de exploración era de hecho distinto de cero y ajustar el valor de un coeficiente de señal significativo para indicar si el coeficiente de exploración era distinto de cero, y por lo tanto significativo. La unidad de codificación de entropía 56 puede entonces determinar si todos los coeficientes de la matriz se han explorado (246). Si no (rama "NO" de 246), la unidad de codificación de entropía 56 puede determinar la ubicación en la matriz que tiene la siguiente probabilidad más alta de inclusión de un coeficiente distinto de cero utilizando la tabla DU (o posiblemente la tabla HDR), y explorar el coeficiente en esta ubicación.

Después de que todos los coeficientes se hayan explorado (rama "Sí" de 246), la unidad de codificación de entropía puede detener la exploración. Además, en algunos ejemplos, la unidad de codificación de entropía 56 puede establecer el valor de una última señal de coeficiente para cada coeficiente para indicar si el coeficiente correspondiente es el último coeficiente en la señal. Después de determinar que todos los coeficientes se han explorado (rama "Sí" de 246), la unidad de codificación de entropía 56 puede establecer el valor del último coeficiente de señal correspondiente al último coeficiente explorado igual a uno. Utilizando las técnicas de la figura 12 como se describe a continuación, la unidad de codificación de entropía 56 puede codificar elementos de sintaxis que incluyen las señales de coeficiente significativo y las últimas señales de coeficiente.

En algunos ejemplos, después de la exploración (ya sea adaptativa o fija), el codificador de vídeo 20 puede ajustar a cero los coeficientes de la matriz producidos por la exploración, por ejemplo, todos los coeficientes después de la posición N de la matriz, donde N es un número entero entre cero y la longitud de la matriz. En otros ejemplos, el codificador de vídeo 20 puede poner a cero los coeficientes en ciertos lugares de la matriz después de la transformación(es) o la cuantificación. Estas ubicaciones pueden corresponder a una esquina superior izquierda de la matriz, por ejemplo. En general, la puesta a cero estos coeficientes puede resultar en la reducción a cero de los coeficientes de frecuencias más altas, lo que puede mejorar la eficiencia de codificación sin mucho impacto en la calidad.

De esta manera, el procedimiento de la figura 11 es un ejemplo de un procedimiento que incluye la transformación de los datos residuales para un bloque de datos de vídeo utilizando una primera transformación para producir una secuencia intermedia bidimensional de coeficientes de transformación, la transformación de la secuencia intermedia bidimensional de coeficientes de transformación usando una transformación rotacional para producir un bloque bidimensional de coeficientes de transformación, la selección de un conjunto de estadísticas asociadas con al menos una de la primera transformación y la transformación de rotación, en el que el conjunto de estadísticas proporciona probabilidades de que las ubicaciones en el bloque bidimensional de coeficientes de transformación serán diferentes de cero, y explorar adaptativamente el bloque bidimensional de coeficientes de transformación basados en el conjunto seleccionado de estadísticas.

La figura 12 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la selección de un modelo de contexto a utilizar cuando se exploran y codifican por entropía los elementos de sintaxis que describen coeficientes adaptativos explorados. Aunque se ha descrito generalmente como realizado por componentes del codificador de vídeo 20 (figura 2) para fines de explicación, debe entenderse que otras unidades de codificación de vídeo, tales como procesadores, unidades de procesamiento, unidades de codificación basadas en hardware, tales como codificadores/decodificadores (CODECs), y similares, también se pueden configurar para realizar el procedimiento de la figura 12.

Debe entenderse también que en otros ejemplos, procedimientos similares pueden incluir medidas adicionales o alternativas a las ilustradas en la figura 12, o pueden realizar las etapas ilustradas en un orden diferente, sin apartarse de las técnicas descritas. Las técnicas para la selección de un modelo de contexto a utilizar en la exploración y la codificación por entropía de elementos de sintaxis que describen coeficientes explorados adaptativamente como se ilustra en la figura 11 pueden corresponder a las etapas 192 a 196 de la figura 8. Las técnicas de la figura 12 pueden realizarse antes, durante o después de que se realice la exploración adaptativa de la figura 11.

La unidad de codificación de entropía 56 puede recibir una matriz de coeficientes de transformación cuantificados (250), por ejemplo, desde la unidad de cuantificación 54. En general, usando el procedimiento de ejemplo de la figura 12, la unidad de codificación de entropía 56 puede codificar elementos de sintaxis que describen los coeficientes recibidos. Los elementos de sintaxis pueden incluir, para cada coeficiente, una señal de coeficiente significativa y una última señal de coeficiente. La señal de coeficiente significativa puede indicar si el coeficiente correspondiente es significativo, por ejemplo, si el valor del coeficiente correspondiente es mayor que cero. La última señal de coeficiente puede indicar si el coeficiente correspondiente es el último coeficiente de una exploración adaptativa.

La unidad de codificación de entropía 56 puede determinar las posiciones de los coeficientes significativos en la matriz recibida. La unidad de codificación de entropía 56 se puede formar elementos de sintaxis que indican las posiciones de los coeficientes significativos en la matriz recibida (252). Por ejemplo, para cada coeficiente de la matriz, la unidad de codificación de entropía 56 puede determinar si el coeficiente es mayor que cero, y si es así, establecer un valor en una matriz de elementos de sintaxis mapeado con el coeficiente igual a uno, de otro modo, la unidad de codificación de entropía puede establecer el valor de mapeo con el coeficiente igual a cero. La unidad de codificación de entropía 56 entonces puede actualizar una tabla de exploración de actualización dinámica utilizando la matriz de elementos de sintaxis (254). Por ejemplo, la unidad de codificación de entropía 56 puede añadir, en el valor actual de cada elemento en la tabla de exploración de actualización dinámica, el valor del elemento de sintaxis yuxtapuesto en la matriz de elementos de sintaxis.

La unidad de codificación de entropía 56 puede entonces explorar un primero de los elementos de sintaxis en la matriz de elementos de sintaxis (256). La unidad de codificación de entropía 56 puede aplicar una exploración en zigzag, tal como la mostrada en la figura 6A, o una exploración seleccionada en base a un tipo de bloque (bloque inter o intra-predicho), una dirección de predicción espacial si el bloque es una bloque codificado de intra-predicción, y/o un tipo de transformación utilizada (por ejemplo, DCT o transformación direccional). A continuación, la unidad de codificación de entropía 56 puede seleccionar un modelo de contexto para la codificación del elemento de sintaxis explorado (258). En general, el modelo de contexto puede seleccionarse basándose en el número de coeficientes significativos (por ejemplo, distintos de cero) de los coeficientes previamente explorados N, donde N es un valor entero distinto de cero. N puede seleccionarse basándose en el tamaño del bloque.

Después de seleccionar el modelo de contexto a utilizar para codificar el elemento de sintaxis actual, la unidad de codificación de entropía 56 puede codificar por entropía el elemento de sintaxis explorado utilizando el modelo de contexto seleccionado (260). La unidad de codificación de entropía 56 puede entonces determinar si el elemento de sintaxis codificado es el último elemento de sintaxis a codificar (262). Si el elemento de sintaxis es el último elemento de sintaxis (rama "SI" de 262), la unidad de codificación de entropía 56 puede detener la exploración de los coeficientes. Por otro lado, si el elemento de sintaxis no es el último elemento de sintaxis (rama "NO" de 262), la unidad de codificación de entropía 56 puede explorar el siguiente elemento de sintaxis (264), y de nuevo seleccionar un modelo de contexto para la codificación de los elementos de sintaxis explorados, por ejemplo, sobre la base de un número de coeficientes significativos en los coeficientes previamente escaneados N.

El ejemplo de la figura 12 se describe principalmente respecto a los elementos de sintaxis que describen si los coeficientes particulares son o no significativos. Estos elementos de sintaxis pueden incluir, por ejemplo, señales de coeficiente significativas, por ejemplo, señales de un bit indicativas de si correspondientes coeficientes son significativos, por ejemplo, distintos de cero. Debe entenderse que técnicas similares se pueden aplicar respecto a los elementos de sintaxis que describen si un coeficiente particular es el último coeficiente en la exploración adaptativa. Por ejemplo, técnicas similares pueden ser aplicadas a una última señal de coeficiente. Cuando se codifican las últimas señales de coeficiente usando CABAC, el modelo de contexto puede estar basado en el índice de orden en la exploración de adaptación que se basa en el tipo de bloque, en la dirección de predicción espacial, y/o en una transformación seleccionada.

Las técnicas de la figura 12 pueden realizarse mediante un dispositivo de codificación de vídeo, tal como el codificador de vídeo 20. Un decodificador de vídeo puede realizar una exploración inversa utilizando los elementos de sintaxis codificados de acuerdo con la figura 12. Por ejemplo, el decodificador de vídeo 30 puede recibir una indicación de un modo de intra-predicción usado para predecir un bloque codificado, una indicación de una rotación de transformación utilizada para transformar el bloque codificado, u otros de tales datos. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse con la misma tabla de alto rango dinámico. En los ejemplos donde el codificador de vídeo 20 incluye varias tablas de alto rango dinámico, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse con el mismo conjunto de tablas de alto rango dinámico. En estos ejemplos, el decodificador de vídeo 30 puede utilizar la información recibida para seleccionar la misma tabla de alto rango dinámico que la utilizada por el codificador de vídeo 20 para realizar la exploración adaptativa.

Como se señaló anteriormente, el codificador de vídeo 20 podrá realizar la exploración adaptativa basada en estadísticas indicativas de la probabilidad de que un coeficiente en una posición particular en una matriz no sea cero. El codificador de vídeo 20 podrá mantener una tabla de estudio de actualización dinámica que indica esta posibilidad, actualizando la tabla de exploración de actualización dinámica para cada bloque explorado. Mediante la codificación de los elementos de sintaxis indicativos de cuál de los coeficientes de un cierto bloque son significativos, y qué coeficiente es el último en la exploración adaptativa, el codificador de vídeo 20 puede proporcionar información al decodificador de vídeo 30 que puede ser usada para invertir los coeficientes de exploración recibidos.

Por ejemplo, el decodificador de vídeo 30 puede decodificar los elementos de sintaxis, a continuación actualizar una versión local de la tabla de exploración de actualización dinámica usando los elementos de sintaxis. El decodificador de vídeo 30 puede entonces decodificar por entropía los coeficientes codificados y colocar los coeficientes decodificados en una posición correspondiente de una matriz que tiene una más alta probabilidad de ser significativo (por ejemplo, distinto de cero). De esta manera, el decodificador de vídeo 30 puede reconstruir una matriz de coeficientes de transformación cuantificados a partir de un vector recibido de los coeficientes codificados por entropía utilizando una exploración inversa adaptativa.

De esta manera, el procedimiento de la figura 12 es un ejemplo de un procedimiento que incluye la exploración de un bloque bidimensional de coeficientes de transformación para producir un vector unidimensional de los coeficientes de transformación, determinando los valores indicativos si los coeficientes de transformación en el vector unidimensional son significativos, y codificando por entropía al menos uno de los valores utilizando un modelo de contexto seleccionado sobre la base de al menos un porcentaje de los coeficientes significativos en un número predeterminado de los valores codificados antes del al menos uno de los valores.

La figura 13 es un diagrama de flujo que ilustra un ejemplo de procedimiento para decodificar una TU que se ha codificado utilizando una o más de las técnicas de esta divulgación. Aunque se ha descrito generalmente como realizado mediante componentes del decodificador de vídeo 30 (figura 3) para fines de explicación, debe entenderse que otras unidades de decodificación de vídeo, tales como procesadores, unidades de procesamiento, unidades de codificación basadas en hardware, tales como codificadores/decodificadores (CODECs), y similares, también se pueden configurar para realizar el procedimiento de la figura 13. También debe entenderse que en otros ejemplos, procedimientos similares pueden incluir etapas adicionales o alternativas a las ilustradas en la figura 13, o pueden realizar las etapas ilustradas en un orden diferente, sin apartarse de las técnicas descritas.

Inicialmente, el decodificador de vídeo 30 podrá recibir datos residuales codificados (300). En el ejemplo de la figura 13, los datos residuales corresponden a la residual de una CU incluyendo uno o más PU predichos en un modo de intra-predicción, para fines de ilustración. De acuerdo con las técnicas de esta divulgación, el decodificador de vídeo 30 puede determinar una primera dirección de predicción para los datos de predicción asociados con los datos residuales recibidos (302). La dirección de predicción puede corresponder a un modo de intra-predicción señalado en la raíz de un árbol cuádruple correspondiente a la CU.

El decodificador de vídeo 30 puede determinar una segunda dirección de predicción mapeada a la primera dirección de predicción (304). Por ejemplo, los datos de mapeo 84 pueden proporcionar un mapeo de muchos a uno de un conjunto de modos de intra-predicción para un conjunto más pequeño, por ejemplo, un subconjunto, de los modos de intra-predicción. En consecuencia, el decodificador de vídeo 30 puede referirse a los datos de mapeo 84 para determinar la segunda dirección de predicción mapeada a la primera dirección de predicción. La unidad de decodificación de entropía 70 del decodificador de vídeo 30 puede entonces comenzar la decodificación por entropía de los coeficientes recibidos (306).

La unidad de decodificación de entropía 70 también puede explorar de manera inversa los coeficientes durante o después de decodificación de entropía (308). En algunos ejemplos, la unidad de decodificación de entropía 70 puede establecer una exploración inversa a la que se mapea la segunda dirección de predicción, por ejemplo, como se indica mediante los datos de mapeo 84. En otros ejemplos, por ejemplo, cuando una primera transformación es en cascada mediante una transformación de rotación, la unidad de decodificación de entropía 70 puede realizar un análisis dinámico inverso. Como se describió anteriormente, la unidad de decodificación de entropía 70 puede, en tales ejemplos, recibir y decodificar los elementos de sintaxis, tales como señales de coeficiente significativas y las últimas señales de coeficientes, tales que la unidad de decodificación de entropía 70 puede producir una tabla de actualización dinámica idéntica a la utilizada mediante un codificador, como el codificador de vídeo 20, cuando el

codificador explora de forma adaptativa los datos residuales.

En cualquier caso, después de la exploración inversa, la unidad de decodificación de entropía 70 puede producir una matriz de dos dimensiones que incluye coeficientes de transformación cuantificados. Por lo tanto, la unidad de cuantificación inversa 76 puede cuantificar los coeficientes de transformación inversa cuantificados de la matriz (310). La unidad de transformación inversa 78 puede seleccionar una transformación inversa mapeada a la segunda dirección de predicción (312) y la transformación inversa de los coeficientes de transformación utilizando la transformación inversa seleccionada (314). Por ejemplo, la unidad de transformación inversa 76 puede referirse a los datos de mapeo 84 para seleccionar la transformación inversa. En algunos ejemplos, los datos de mapeo 84 pueden indicar una transformación de rotación inversa y otra transformación inversa a aplicar, en cuyo caso la unidad de transformación inversa 78 puede aplicar primero la transformación de rotación inversa y a continuación aplicar la otra transformación inversa a los coeficientes de transformación.

De acuerdo con las técnicas de esta divulgación, en algunos ejemplos, puede haber un sub-bloque de los coeficientes de transformación para que múltiples transformaciones inversas sean posibles. En tales ejemplos, el decodificador de vídeo 30 puede determinar la transformación inversa a aplicar utilizando una indicación de la transformación aplicada por el codificador de vídeo. Por ejemplo, el decodificador de vídeo 30 puede recibir una indicación de la transformación utilizada para el sub-bloque en el nodo raíz de un árbol cuádruple correspondiente al bloque que incluye los datos residuales.

Después de aplicar la(s) transformación(es) inversa(s), el decodificador de vídeo 30 obtiene bloques residuales similares a los calculados por el codificador de vídeo 20 durante la codificación de los datos de vídeo. La unidad de intra-predicción 74 puede proporcionar una unidad de predicción de s datos residuales al sumador 80, que puede combinar la unidad de predicción y los datos residuales para producir una CU decodificada (316). El decodificador de vídeo 30 puede montar un fotograma decodificado incluyendo la CU decodificada en la memoria de fotogramas de referencia 82. El fotograma decodificado puede entonces transformarse para mostrarse y/o utilizarse como referencia para la decodificación de otros fotogramas.

De esta manera, el procedimiento de la figura 13 es un ejemplo de un procedimiento que incluye la recepción de una indicación de un primer modo de intra-predicción en un primer conjunto de modos de intra-predicción para un bloque de datos de vídeo codificados, la determinación de un segundo modo de intra-predicción a partir de un conjunto más pequeño de modos de intra-predicción en los que se mapea el primer modo de intra-predicción, la determinación de una transformación direccional inversa a la que se mapea el segundo modo de intra-predicción, y la aplicación de la transformación direccional inversa en datos transformados residuales del bloque.

El procedimiento de la figura 13 es también un ejemplo de un procedimiento que incluye recibir una primera indicación de un modo de intra-predicción a utilizar para decodificar un bloque de datos de vídeo, determinar si el bloque incluye un sub-bloque de un tamaño para el que varias transformaciones son posibles en base al tamaño del sub-bloque y el modo de intra-predicción indicado, cuando el bloque incluye el sub-bloque del tamaño para que varias transformaciones inversas sean posibles en función del tamaño del sub-bloque y el modo de intra-predicción indicado, recibiendo una segunda indicación de una de las múltiples posibles transformaciones inversas, y transformar de manera inversa el sub-bloque usando la indicada de las múltiples transformaciones inversas posibles.

El procedimiento de la figura 13 es también un ejemplo de un procedimiento que incluye la recepción de una indicación de que los datos residuales para un bloque de datos de vídeo se transformó utilizando una primera transformación y una transformación de rotación para producir un bloque bidimensional de coeficientes de transformación, la selección de un conjunto de estadísticas asociadas con al menos una de la primera transformación y la transformación de rotación, en el que el conjunto de estadísticas proporciona probabilidades de que las ubicaciones en el bloque bidimensional de coeficientes de transformación será distinto de cero, y de explorar de manera adaptativa inversa un vector unidimensional recibido que incluye una versión codificada de los datos residuales para el bloque basado en el conjunto de estadísticas seleccionado para producir una matriz bidimensional de coeficientes de transformación para el bloque.

El procedimiento de la figura 13 es también un ejemplo de un procedimiento que incluye la recepción de valores indicativos de si los coeficientes de transformación codificados en un vector unidimensional recibido de los coeficientes de transformación codificados son significativos, la decodificación por entropía de al menos uno de los valores utilizando un modelo de contexto seleccionado sobre la base de al menos un porcentaje de coeficientes significativos en un número predeterminado de los valores decodificados antes del al menos uno de los valores, y explorar de manera inversa el vector unidimensional para producir un bloque de dos dimensiones de los coeficientes de transformación.

La figura 14 es un diagrama de flujo que ilustra un ejemplo de procedimiento para la selección de una transformación para aplicarse a un bloque intra-codificado que incluye un borde para el que se indica el modo de intra-predicción DC. Aunque se ha descrito respecto al decodificador de vídeo 30 de la figura 3, se debe entender que técnicas similares (recíprocas) pueden ser aplicadas mediante el codificador de vídeo 20 de la figura 2, u otros dispositivos de codificación de vídeo.

El codificador de vídeo 30 puede recibir un bloque intra-codificado, por ejemplo, una TU (180). El bloque puede comprender un bloque de coeficientes de transformación correspondientes a un nodo en un árbol cuádruple de TU. El nodo de árbol cuádruple de TU puede incluir una indicación del modo de intra-predicción que debe aplicarse para calcular un valor de predicción para el bloque. En consecuencia, el decodificador vídeo 30 puede determinar el modo de predicción y si el modo de predicción DC está señalado para el bloque (352). Si el modo de predicción DC está señalado para el bloque (rama "SI" de 352), el decodificador de vídeo 30 puede determinar además si existe una ventaja en el bloque (354). Por ejemplo, como se describió anteriormente, el decodificador de vídeo 30 puede examinar los bloques vecinos previamente codificados para determinar si se detecta un borde en los bloques previamente codificados, y si el borde interseca un límite entre el bloque previamente codificado y el bloque actual.

Si se determina que existe un borde en el bloque (rama "Sí" de 354), el decodificador de vídeo 30 podrá calcular un valor predicho para el bloque utilizando un modo de predicción basado en bordes (356). Además, el decodificador de vídeo 30 puede determinar un ángulo del borde (358) y determinar un intra-modo con un ángulo que se aproxima mucho al ángulo del borde (360). Por ejemplo, el decodificador de vídeo 30 puede calcular las diferencias entre ángulos para uno o más de los posibles modos de intra-predicción y el ángulo del borde, y seleccionar el modo de intra-predicción que tiene la diferencia más baja.

La determinación de este modo de predicción se suele realizar solamente para determinar la transformación que se mapea a ese modo de predicción, aunque como el decodificador de vídeo 30 generalmente predice un valor para el bloque con el modo de predicción basado en bordes, en este ejemplo. Es decir, el decodificador de vídeo 30 a continuación puede seleccionar una transformación mapeada al modo de intra-predicción determinado (362), es decir, el modo de intra-predicción que tiene un ángulo que se aproxima mucho al ángulo del borde. El decodificador de vídeo 30 puede entonces transformar el bloque usando la transformación seleccionada (364).

Por otro lado, si el modo de predicción DC no se señaló para el bloque (rama "NO" de 352), el decodificador de vídeo 30 puede predecir el bloque usando el modo señalado (353). Si un borde no está determinado a existir en el bloque cuando se señala el modo de predicción DC (rama "NO" de 354), el decodificador de vídeo 30 puede predecir el bloque usando modo de predicción DC, como señalado (366). El decodificador de vídeo 30 también puede seleccionar la transformación mapeada al modo de predicción (por ejemplo, DC o direccional, como señalado) (368) o, en algunos ejemplos una transformación por defecto, tal como una DCT. El decodificador vídeo 30 también puede transformar el bloque con la transformación seleccionada en este caso (364).

Después de transformar el bloque, que en este ejemplo corresponde a la transformación inversa del bloque, el decodificador de vídeo 30 reproduce un bloque de valores residuales en el dominio espacial. Para decodificar el bloque, el decodificador de vídeo 30 puede agregar el bloque de valores residuales al bloque predicho (resultante de la etapa 353, 356, ó 366). Las etapas de añadir el valor residual al valor predicho no se muestran en la figura 14 por razones de brevedad, pero puede realizarse después de la etapa 364.

De esta manera, el procedimiento de la figura 14 es un ejemplo de un procedimiento que incluye la recepción de información que indica que un modo de intra-predicción para un bloque de datos de vídeo es un modo de intra-predicción DC, la determinación de un ángulo de un borde en el bloque de datos de vídeo basado en la indicación del modo de intra-predicción DC para el bloque, la transformación inversa del bloque utilizando una transformación inversa de dirección mapeada a un modo de intra-predicción direccional que tiene un ángulo que se aproxima al ángulo del borde, y la decodificación del bloque de transformación inversa.

Como se señaló anteriormente, un procedimiento similar puede realizarse mediante, por ejemplo, el codificador de vídeo 20. Tal procedimiento puede incluir la determinación de que un bloque codificado con intra-predicción contiene un borde dentro del bloque, el cálculo de un bloque residual para el bloque basado en un valor de predicción calculado utilizando un modo de intra-predicción dirigido a bordes, la transformación del bloque residual utilizando una transformación direccional mapeada a un modo de intra-predicción direccional que tiene un ángulo que se aproxima a un ángulo del borde, y el envío de información representativa del bloque residual transformado e información que indica que el bloque se predijo usando un modo de intra-predicción DC.

En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware, o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse en o transmitirse a través de una o más instrucciones o código, un medio legible por ordenador y ejecutarse mediante una unidad de procesamiento basado en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tal como un medio de almacenamiento de datos, o medios 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 generalmente pueden corresponder a (1) medios de almacenamiento tangibles legibles por ordenador que no son transitorios o (2) un medio tal como una onda de la señal o portador. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se puede acceder mediante uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la aplicación de las técnicas descritas en esta divulgación. Un producto de programa de ordenador puede incluir un medio legible por ordenador.

A modo de ejemplo, y no de limitación, tales 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 utilizar para almacenar el código del programa deseado en forma de instrucciones o estructuras de datos y que se puede acceder mediante un ordenador. Además, cualquier conexión se denomina correctamente un medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una página web, el servidor o la otra fuente remota mediante un cable coaxial, cable de fibra óptica, par trenzado, línea de abonado digital (DSL), o tecnologías inalámbricas como infrarrojos, radio y microondas, entonces el cable coaxial, cable de fibra óptica, par trenzado, DSL o tecnologías inalámbricas como infrarrojos, radio y microondas están incluidos en la definición de medio. Se debe entender, sin embargo, 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 se dirigen a medios de almacenamiento tangibles no transitorios. Disco, tal como se usa aquí, incluye discos compactos (CD), discos láser, discos ópticos, discos versátiles digitales (DVD), disquetes y discos Blu-ray, donde generalmente se reproducen discos de datos magnéticamente, mientras que los discos reproduzcan datos ópticamente con láser. Combinaciones de los anteriores también deben incluirse dentro del alcance de los medios legibles por ordenador.

Las instrucciones pueden ser ejecutadas mediante uno o más procesadores, tales como uno o más procesadores digitales de señal (DSPs), microprocesadores de propósito general, circuitos integrados de aplicación específica (ASICs), matrices lógicas programables de campo (FPGAs), u otro circuito lógico discreto equivalente o integrado. Por consiguiente, el término "procesador", como se usa en el presente documento puede referirse a cualquiera de la estructura anterior o cualquier otra estructura adecuada para la aplicación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en este documento puede proporcionarse dentro de módulos de hardware y/o software dedicados configurados para la codificación y la decodificación, o incorporados en un códec combinado. Además, las técnicas pueden aplicarse plenamente en uno o más circuitos o elementos lógicos.

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 circuitos integrados (por ejemplo, un conjunto de chips). Varios componentes, módulos o unidades se describen en esta divulgación para enfatizar los aspectos funcionales de los dispositivos configurados para realizar las técnicas descritas, pero no necesariamente requieren la realización de diferentes unidades de hardware. Por el contrario, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionados mediante una colección de unidades de hardware operativas entre sí, incluyendo uno o más procesadores como se describe anteriormente, en conjunción con el software y/o el firmware adecuado.

REIVINDICACIONES

1. Un procedimiento de decodificación de datos de vídeo, comprendiendo el procedimiento:

recibir (201) una primera indicación de un modo de intra-predicción a utilizar para decodificar un bloque de datos de vídeo;

determinar (212) si el bloque incluye un sub-bloque para el que son posibles múltiples transformaciones, en el que la determinación está basada en el tamaño del sub-bloque y el modo de intra-predicción indicado; cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones inversas:

recibir (214) una segunda indicación de una de las múltiples posibles transformaciones inversas; y transformar de manera inversa (216) el sub-bloque utilizando la indicada de las múltiples transformaciones inversas posibles.

2. El procedimiento de la reivindicación 1, que comprende además, cuando el bloque no incluye el sub-bloque para el que son posibles múltiples transformaciones inversas, la transformación inversa (218) del bloque utilizando una transformación inversa asociada con el modo de intra-predicción indicado para el bloque.

3. El procedimiento de la reivindicación 1, que comprende además la transformación inversa (216) de todos los sub-bloques del bloque que tienen el tamaño para el que son posibles múltiples transformaciones inversas utilizando la indicada de las múltiples transformaciones posibles.

4. El procedimiento de la reivindicación 1, en el que el modo de intra-predicción comprende un primer modo de intra-predicción, y en el que el sub-bloque comprende un primer sub-bloque, comprendiendo además el procedimiento:

determinar (204) si el bloque incluye un segundo sub-bloque de un tamaño para el que el primer modo de intra-predicción no está disponible; cuando el bloque incluye el segundo sub-bloque del tamaño para el que el primer modo de intra-predicción no está disponible:

determinar (206) un segundo modo de intra-predicción al que se mapea el primer modo de intra-predicción; y predecir (206) el segundo sub-bloque utilizando el segundo modo de intra-predicción.

5. El procedimiento de la reivindicación 4, que comprende además, cuando el primer modo de intra-predicción está disponible para todos los sub-bloques del bloque, la predicción (208) de todos los sub-bloques utilizando el primer modo de intra-predicción.

6. El procedimiento de la reivindicación 1, en el que la recepción de la primera indicación comprende la recepción de la primera indicación en un nodo raíz de una estructura de datos de árbol cuádruple correspondiente al bloque de datos de vídeo.

7. Un aparato para decodificar datos de vídeo, comprendiendo el aparato:

medios para recibir una primera indicación de un modo de intra-predicción a utilizar para decodificar un bloque de datos de vídeo;

medios para determinar si el bloque incluye un sub-bloque para el que son posibles múltiples transformaciones, en el que la determinación está basada en el tamaño del sub-bloque y el modo de intra-predicción indicado;

medios para recibir una segunda indicación de una de las múltiples posibles transformaciones inversas cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones inversas; y

medios para transformar de manera inversa el sub-bloque utilizando la indicada de las múltiples posibles transformaciones inversas cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones inversas.

8. El aparato de la reivindicación 7, en el que los medios para recibir una primera indicación, los medios para determinar, los medios para recibir una segunda indicación, y los medios para la transformación inversa están incorporados en un decodificador de vídeo.

9. Un producto de programa de ordenador que comprende un medio de almacenamiento legible por ordenador que tiene almacenado en el mismo instrucciones que, al ejecutarse, hacen que un procesador realice el procedimiento de una cualquiera de las reivindicaciones 1 a 6.

10. Un procedimiento de codificación de datos de vídeo, comprendiendo el procedimiento:

seleccionar (201) un modo de intra-predicción a utilizar para codificar un bloque de datos de vídeo;

determinar (212) si el bloque incluye un sub-bloque para el que son posibles múltiples transformaciones, en el que la determinación se basa en el tamaño del sub-bloque y en el modo de intra-predicción seleccionado; cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones:

seleccionar (214) una de las múltiples transformaciones posibles;

transformar (216) el sub-bloque utilizando la seleccionada de las múltiples transformaciones posibles; y proporcionar (214) una indicación de la seleccionada de las múltiples transformaciones posibles para el tamaño del sub-bloque.

5 11. El procedimiento de la reivindicación 10, en el que el modo de intra-predicción comprende un primer modo de intra-predicción, y en el que el sub-bloque comprende un primer sub-bloque, comprendiendo el procedimiento además:

10 determinar (204) si el bloque incluye un segundo sub-bloque de un tamaño para el que el primer modo de intra-predicción no está disponible;
cuando el bloque incluye el segundo sub-bloque del tamaño para el que el primer modo de intra-predicción no está disponible:

determinar (206) un segundo modo de intra-predicción al que se mapea el primer modo de intra-predicción; y predecir (206) el segundo sub-bloque utilizando el segundo modo de intra-predicción.

15 12. El procedimiento de la reivindicación 10, en el que proporcionar la indicación comprende proporcionar la indicación en un nodo raíz de una estructura de datos de árbol cuádruple correspondiente al bloque de datos de vídeo.

13. Un aparato para codificar datos de vídeo, comprendiendo el aparato:

20 medios para seleccionar un modo de intra-predicción a utilizar para codificar un bloque de datos de vídeo;
medios para determinar si el bloque incluye un sub-bloque para el que son posibles múltiples transformaciones, en el que la determinación se basa en el tamaño del sub-bloque y el modo de intra-predicción seleccionado;
medios para seleccionar una de las múltiples transformaciones posibles cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones;
medios para transformar el sub-bloque utilizando la seleccionada de las múltiples transformaciones posibles cuando el bloque incluye el sub-bloque para el que son posibles múltiples transformaciones; y
25 medios para proporcionar una indicación de la seleccionada de las múltiples transformaciones posibles para el tamaño del sub-bloque cuando el bloque incluye el sub-bloque para el que varias transformaciones son posibles.

14. El aparato de la reivindicación 13, en el que los medios para seleccionar un modo de intra-predicción, los medios para determinar, los medios para seleccionar una de las múltiples transformaciones posibles, los medios para transformar, y los medios para proporcionar están incorporados en un codificador de vídeo.

30 15. Un producto de programa de ordenador que comprende un medio de almacenamiento legible por ordenador que tiene almacenado en el mismo instrucciones que, al ejecutarse, hacen que un procesador realice el procedimiento de una cualquiera de las reivindicaciones 10 a 12.

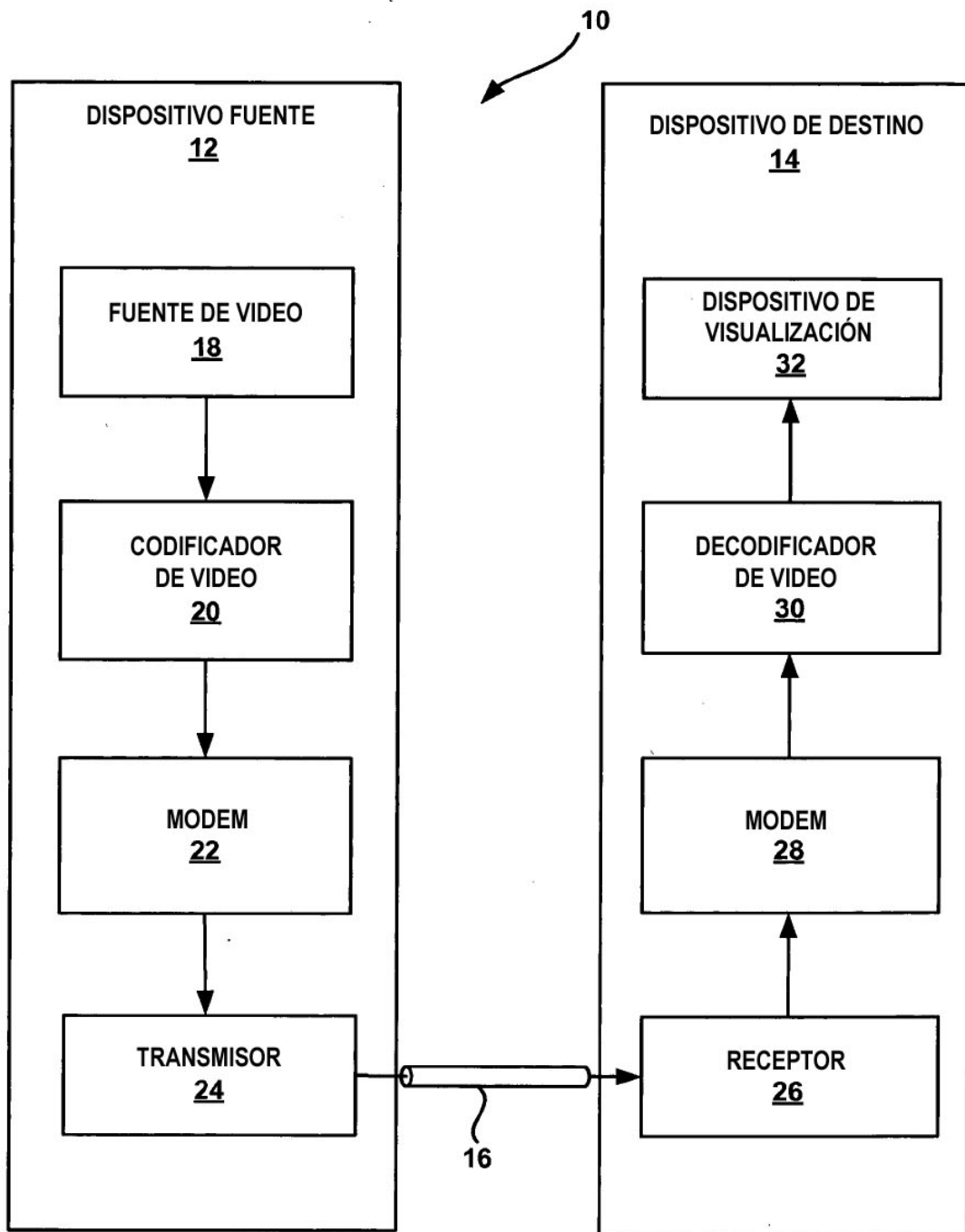


FIG. 1

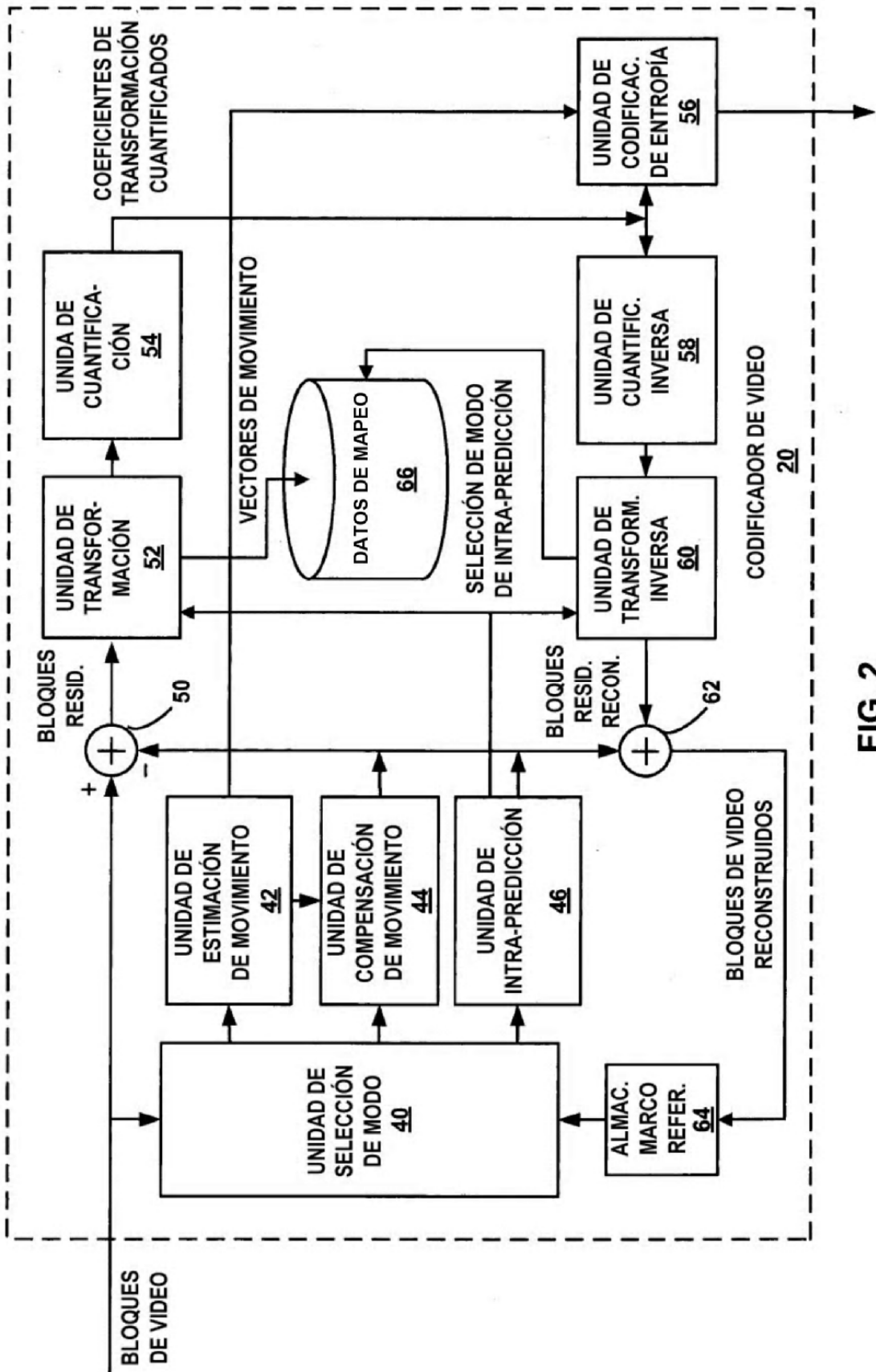


FIG. 2

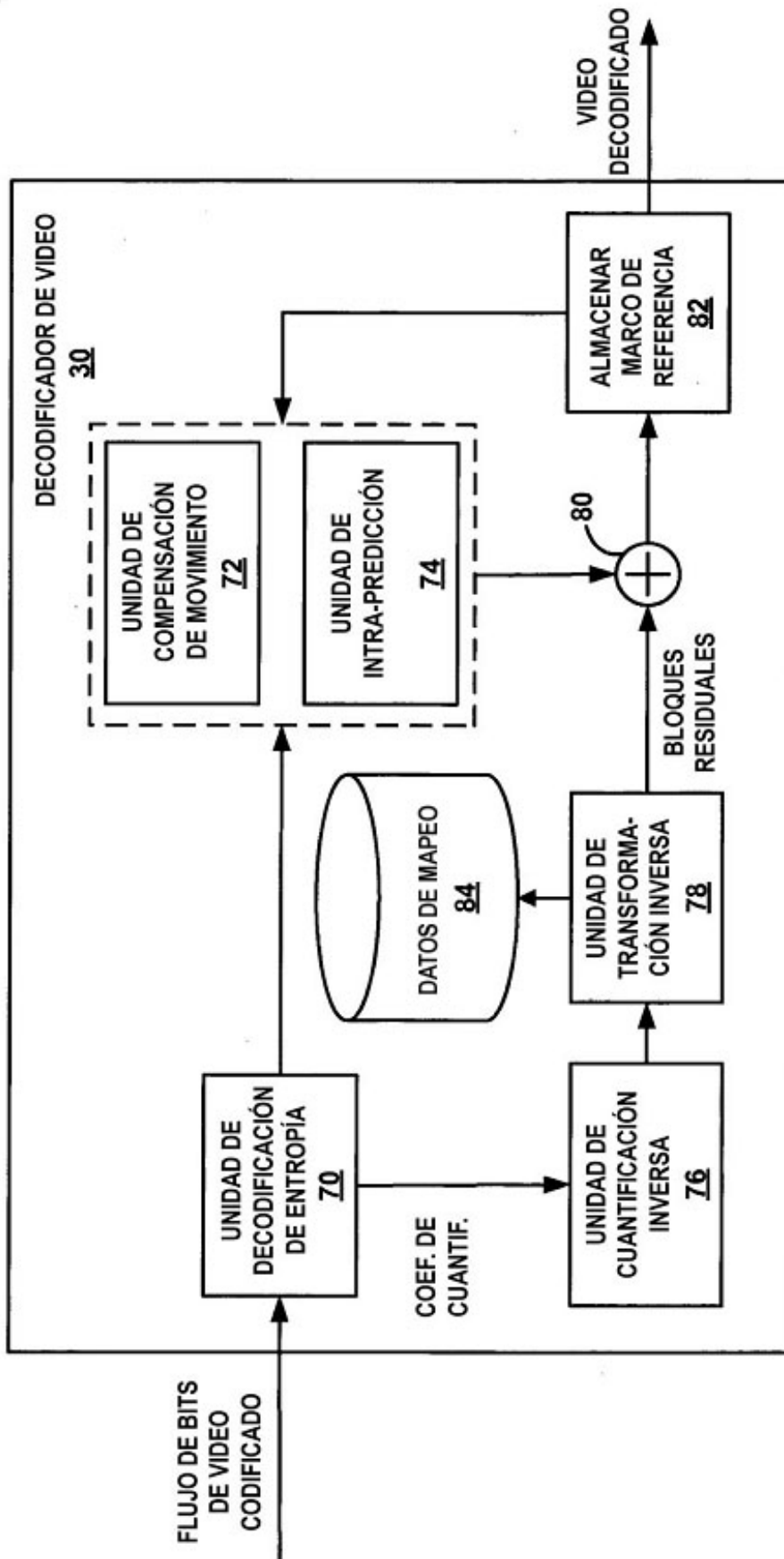


FIG. 3

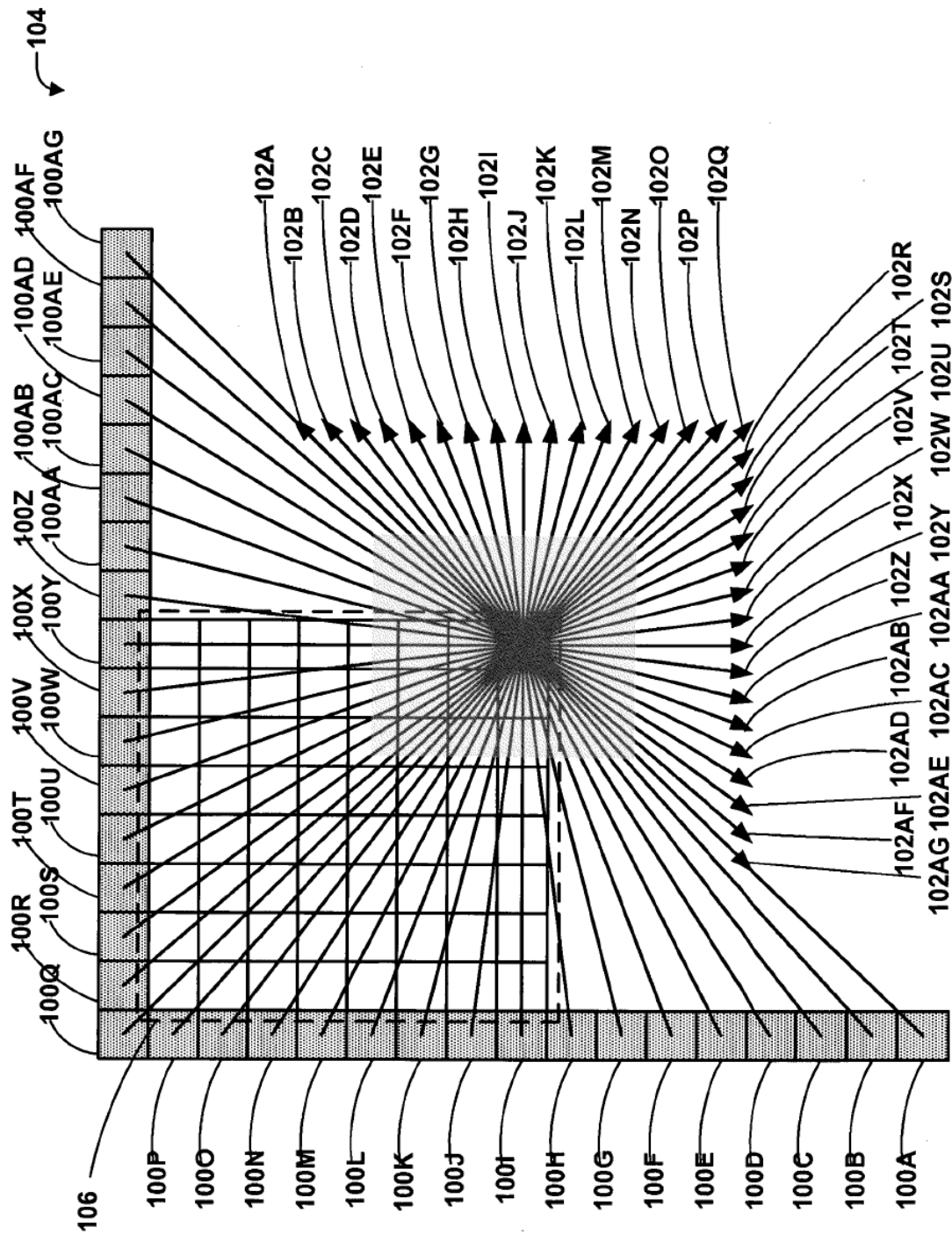


FIG. 4

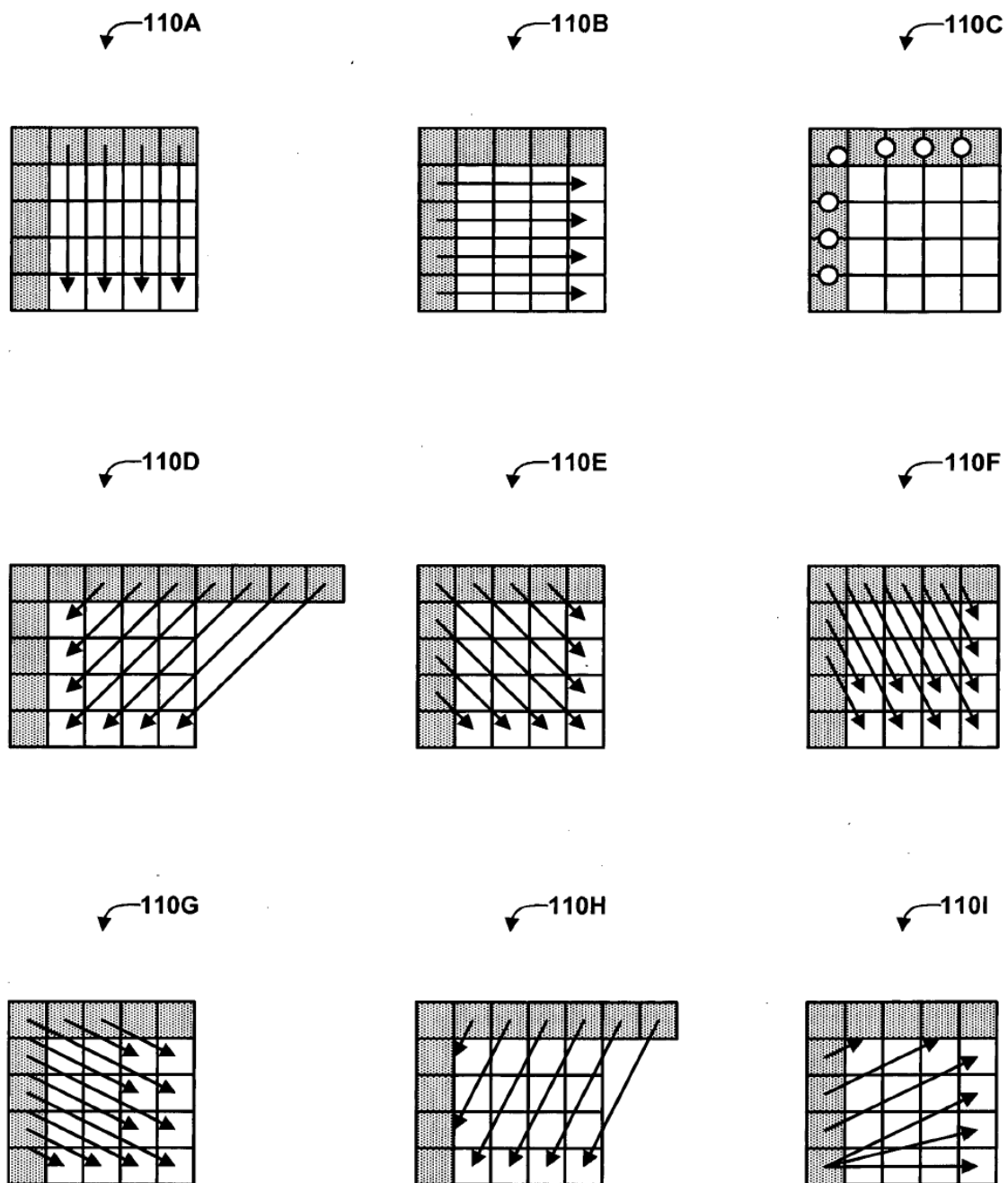


FIG. 5

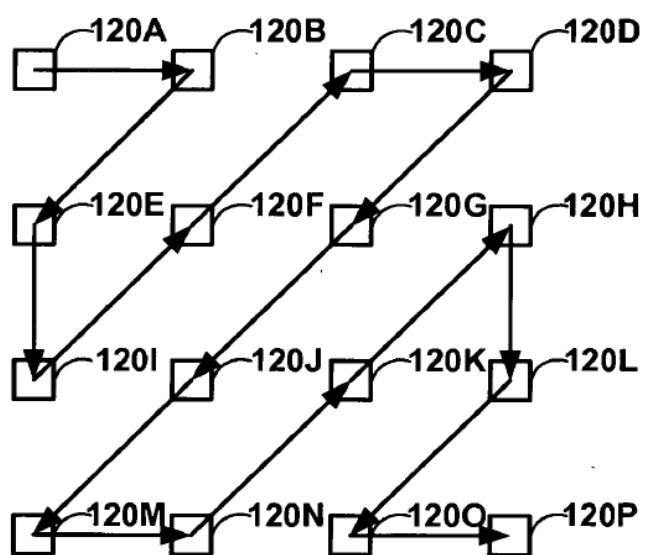


FIG. 6

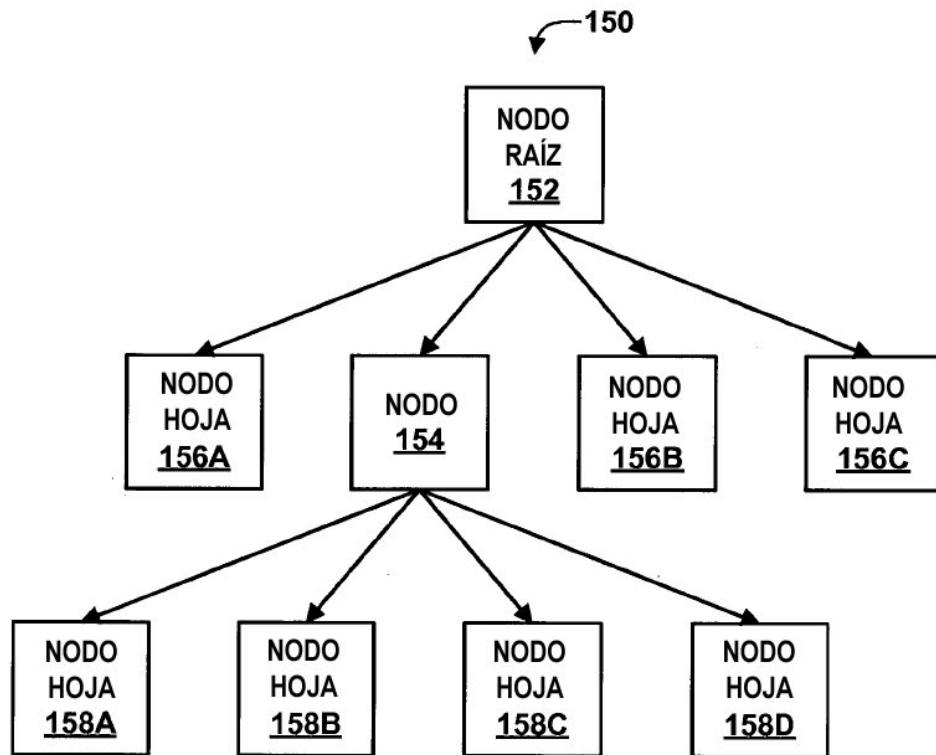


FIG. 7A

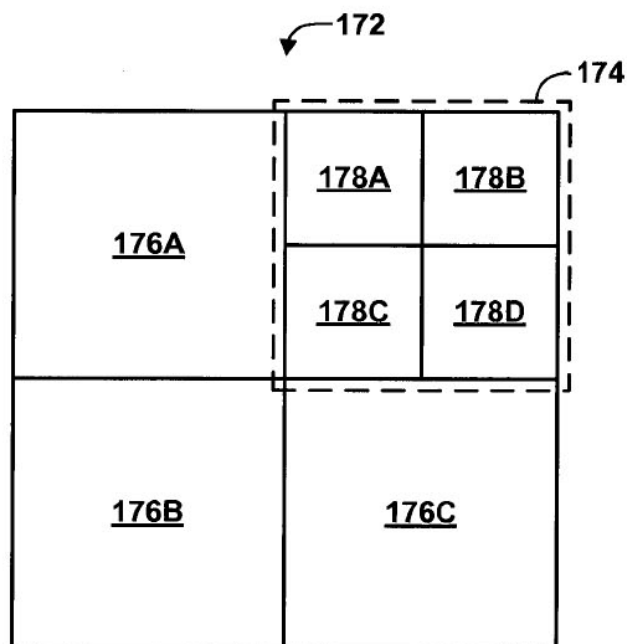


FIG. 7B

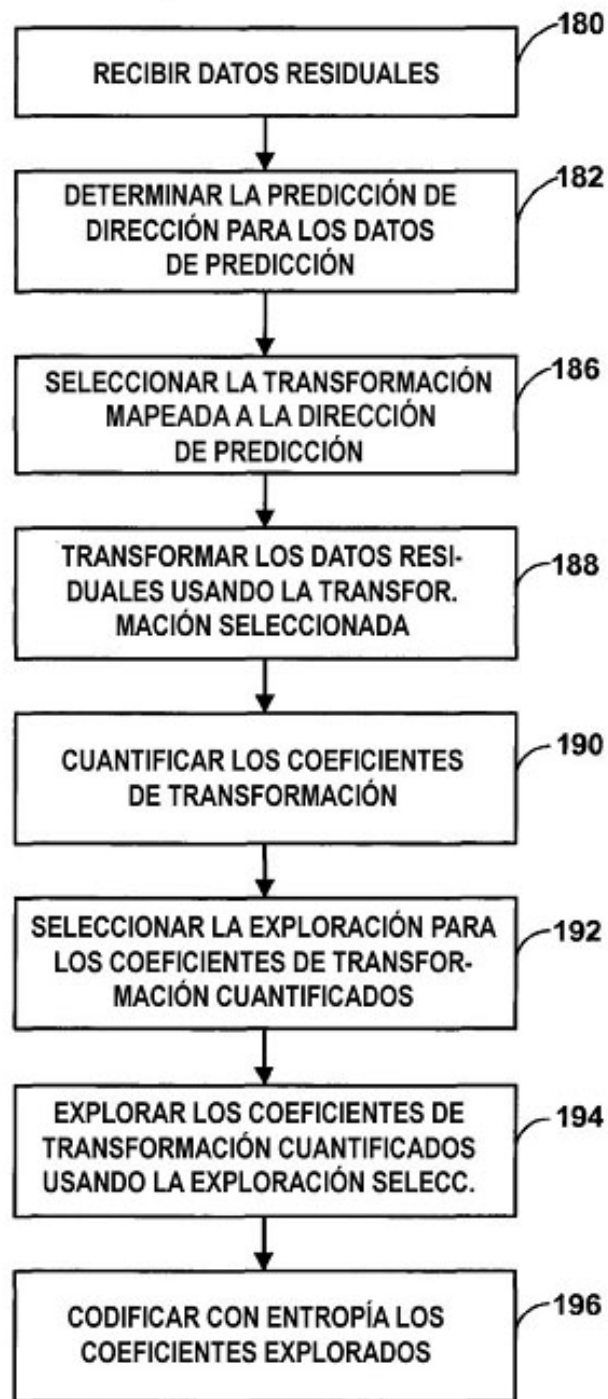


FIG. 8

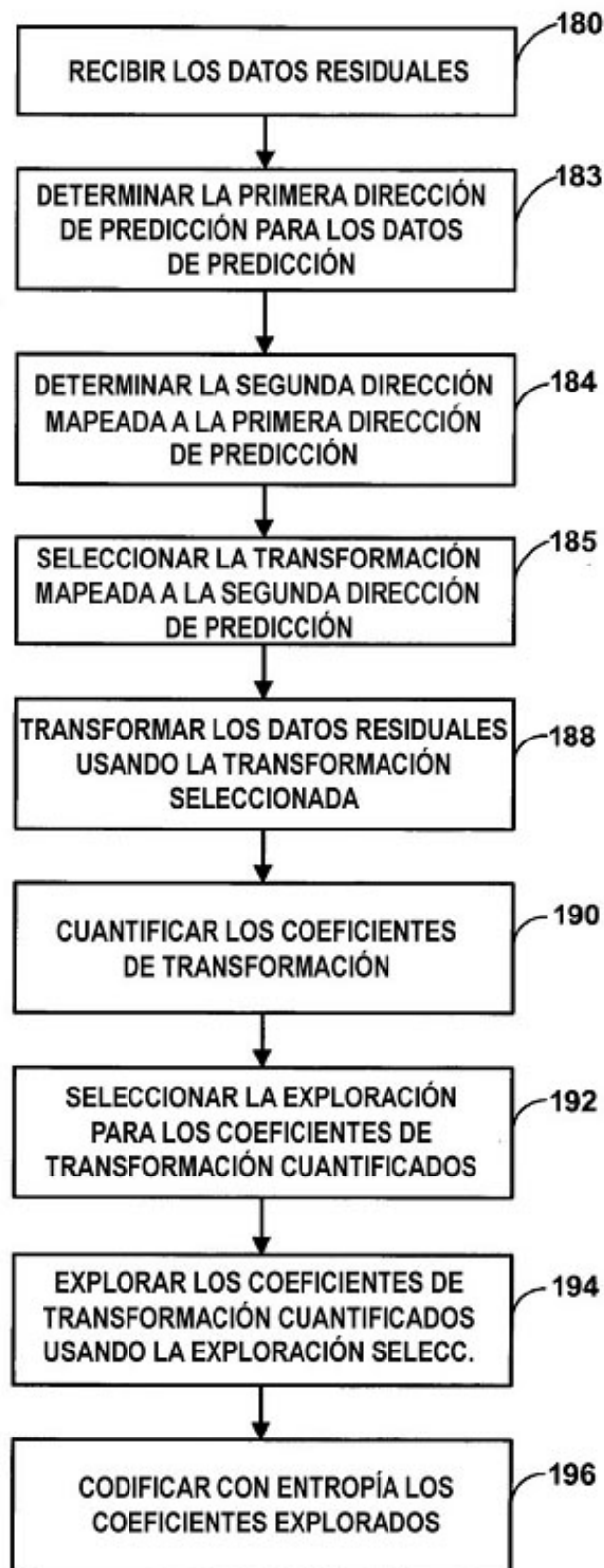


FIG. 9

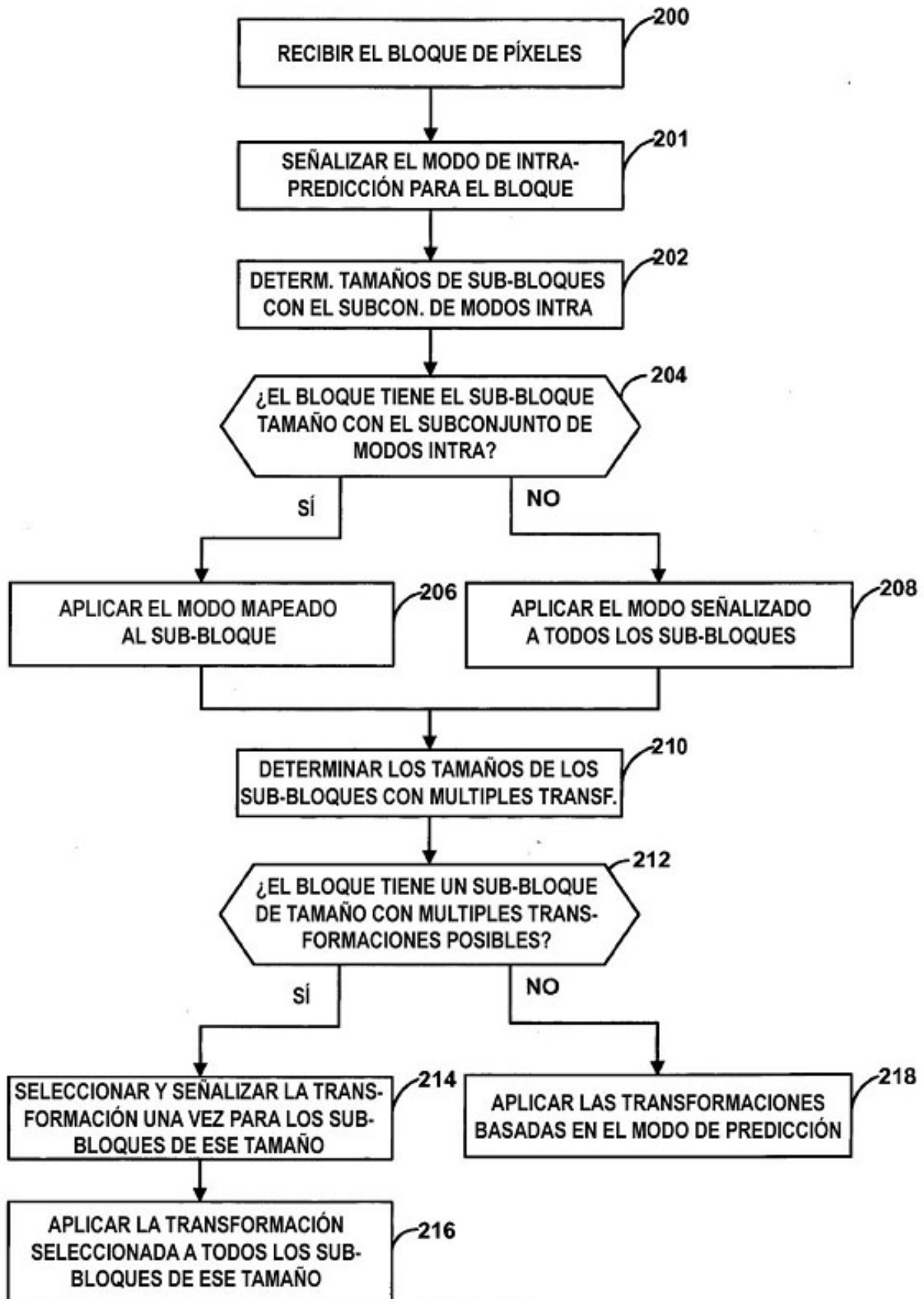


FIG. 10

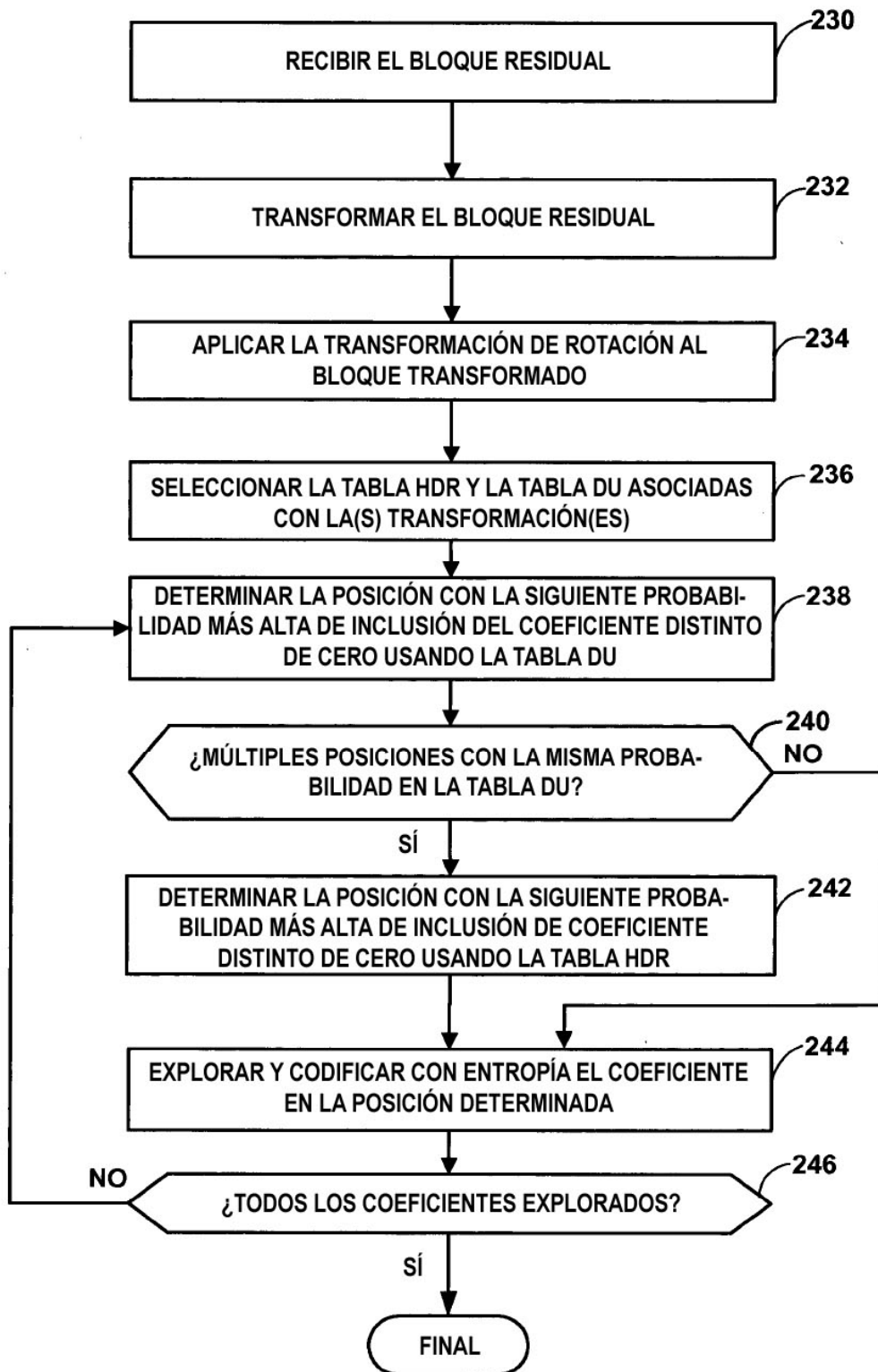


FIG. 11

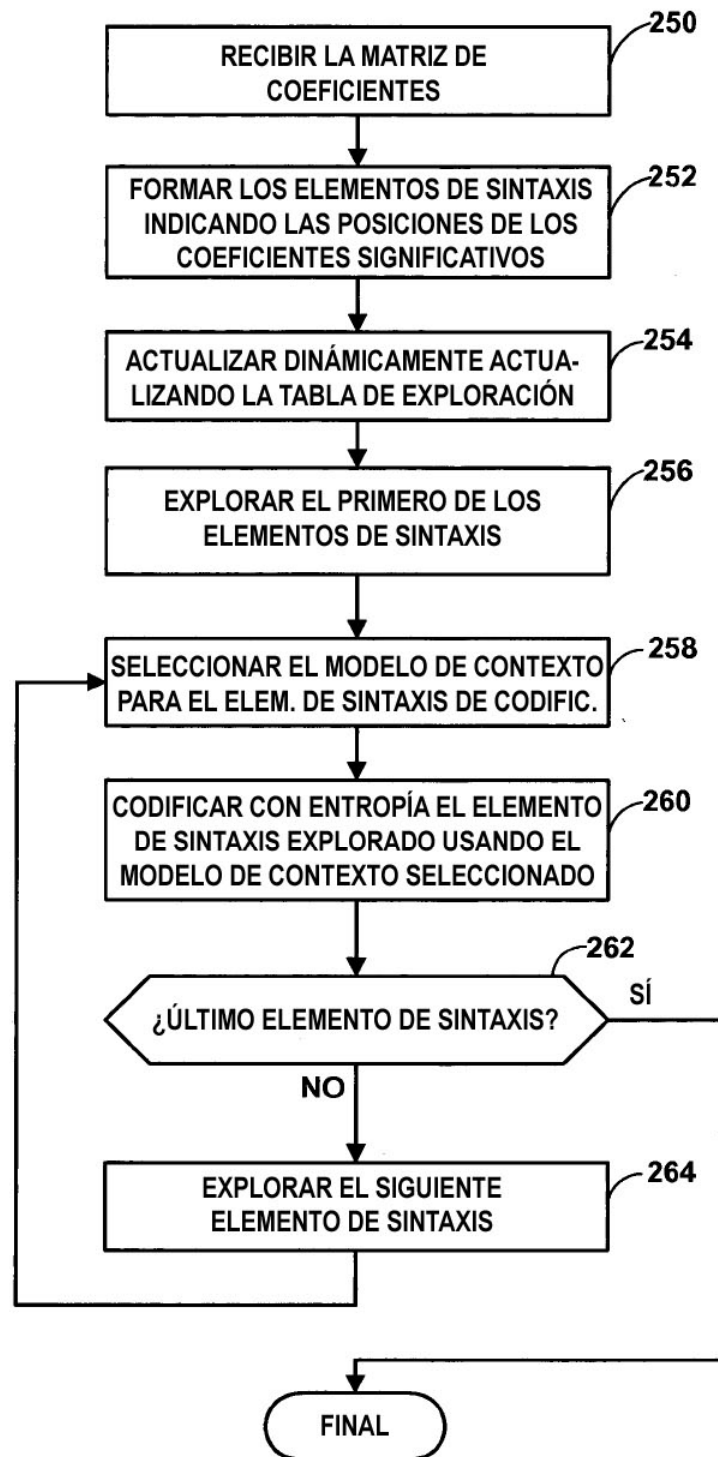


FIG. 12

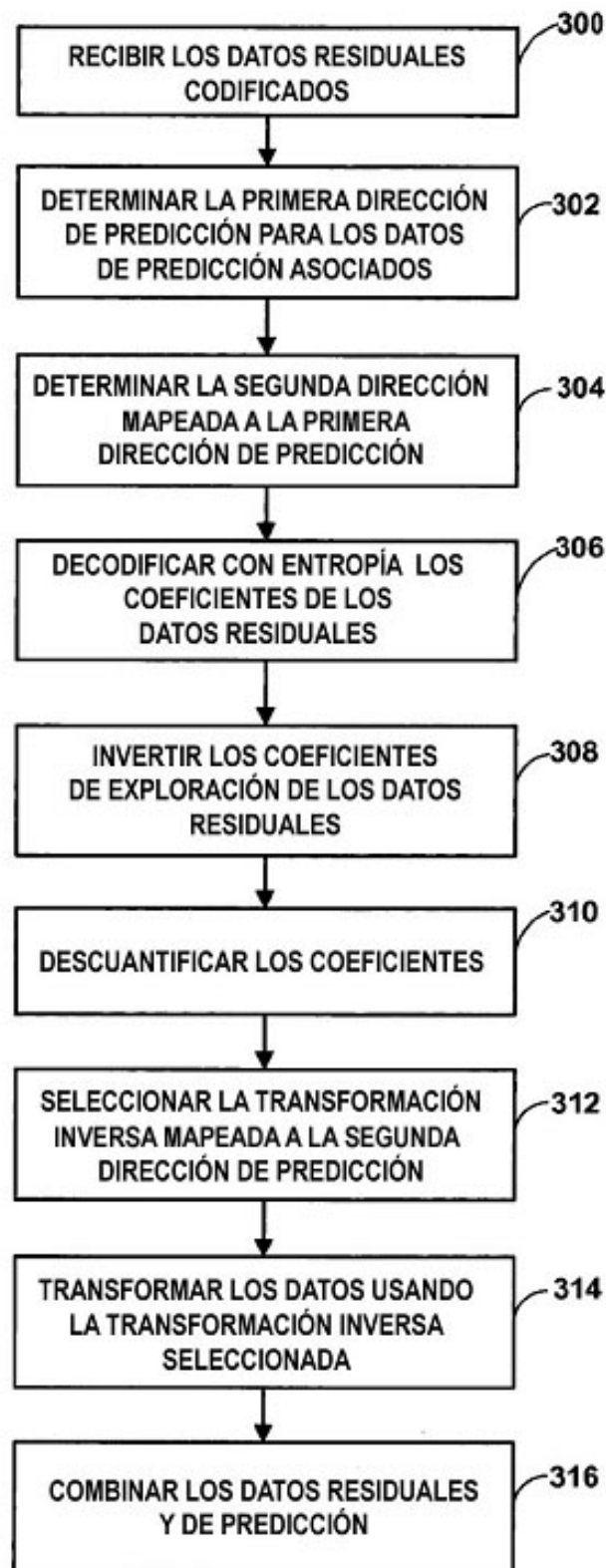


FIG. 13

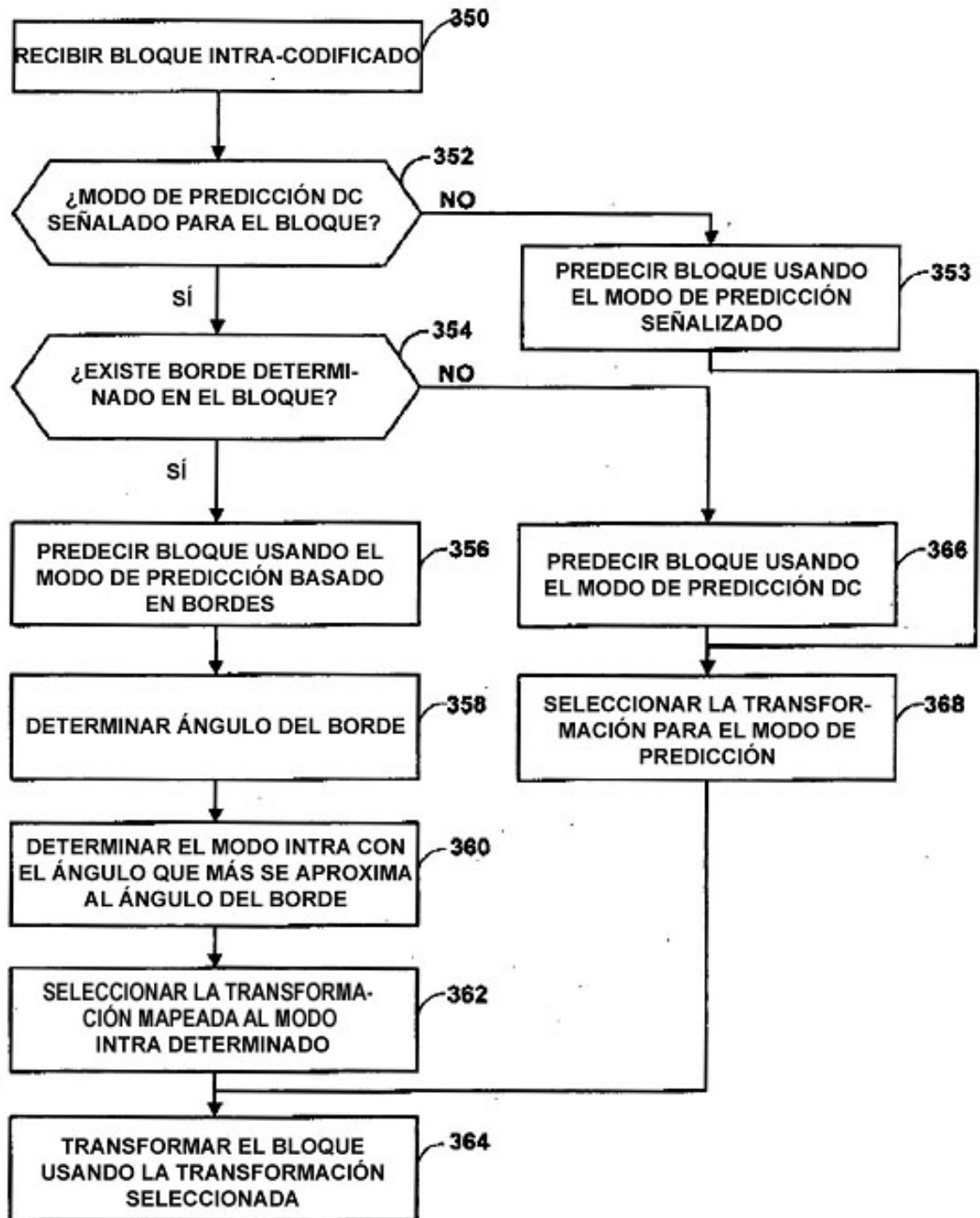


FIG. 14