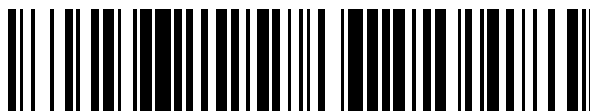


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 727 635**

51 Int. Cl.:

H04N 19/93 (2014.01)

H04N 19/90 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **04.04.2014** **PCT/US2014/033019**

87 Fecha y número de publicación internacional: **09.10.2014** **WO14165789**

96 Fecha de presentación y número de la solicitud europea: **04.04.2014** **E 14720462 (2)**

97 Fecha y número de publicación de la concesión europea: **27.02.2019** **EP 2982126**

54 Título: **Determinación de paletas en la codificación de vídeo basada en paletas**

30 Prioridad:

05.04.2013 US 201361809236 P

10.04.2013 US 201361810649 P

03.04.2014 US 201414244688

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
17.10.2019

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)

5775 Morehouse Drive

San Diego, US

72 Inventor/es:

GUO, LIWEI;

KARCZEWICZ, MARTA;

SOLE ROJALS, JOEL y

JOSHI, RAJAN LAXMAN

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 727 635 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

Determinación de paletas en la codificación de vídeo basada en paletas

5 CAMPO TÉCNICO

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

ANTECEDENTES

10 [0002] Las capacidades de vídeo digital pueden incorporarse a una amplia gama de dispositivos, que incluye televisores digitales, sistemas de radiodifusión digital directa, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, 15 teléfonos celulares o de radio por satélite, los denominados «teléfonos inteligentes», dispositivos de videoconferencia, dispositivos de transmisión de vídeo en tiempo real y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, codificación de vídeo avanzada (AVC), la norma de codificación de vídeo de alta eficiencia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, 20 recibir, codificar, descodificar y/o almacenar información de vídeo digital con más eficiencia implementando dichas técnicas de compresión de vídeo.

[0003] Las técnicas de compresión de vídeo realizan predicción espacial (dentro de las imágenes) y/o predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca a las secuencias de vídeo. Para la codificación 25 de vídeo basada en bloques, un fragmento de vídeo (es decir, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen o la predicción temporal con respecto a muestras de 30 referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas y las imágenes de referencia pueden denominarse tramas de referencia.

[0004] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se va a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque 35 inter-codificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y con los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de píxel en un dominio de transformada, dando como resultado coeficientes residuales, que pueden cuantificarse entonces. Los coeficientes cuantificados, 40 inicialmente dispuestos en una matriz bidimensional, pueden escanearse con el fin de producir un vector unidimensional de coeficientes, y puede aplicarse la codificación por entropía para lograr incluso más compresión.

[0005] Un flujo de bits de codificación multivistas puede generarse codificando vistas, por ejemplo, desde múltiples perspectivas. Se han desarrollado algunas normas de vídeo tridimensional (3D) que hacen uso de aspectos de codificación 45 de múltiples vistas. Por ejemplo, diferentes visualizaciones pueden transmitir visualizaciones de ojo izquierdo y derecho para soportar vídeo 3D. De forma alternativa, algunos procesos de codificación de vídeo 3D pueden aplicar la denominada codificación de múltiples visualizaciones con profundidad. En la codificación de múltiples visualizaciones más profundidad, un flujo de bits de vídeo 3D puede contener no solo componentes de visualización de textura, sino también componentes de visualización de profundidad. Por ejemplo, cada visualización puede comprender un componente de visualización de 50 textura y un componente de visualización de profundidad.

[0006] En el documento US 2011/0110416 A1 de Bally Gaming Inc., un sistema de codificador y decodificador divide una imagen en bloques espaciales, eligiendo un tipo de codificación para cada bloque de una lista de posibles esquemas de codificación. Los esquemas están diseñados individualmente de modo que los procedimientos de decodificación sean 55 una serie de operaciones repetitivas sobre símbolos alineados por octetos en un flujo de datos de longitud fija. El sistema de motor de códec de vídeo habilita una función de búsqueda que avanza en el flujo sin decodificar todas las tramas entre una trama de búsqueda inicial y una trama de búsqueda final. El sistema de motor de códec de vídeo permite la mezcla

alfa de su almacén temporal de salida con una salida desde otro almacén temporal. El sistema presta soporte tanto a tramas fijas como a vídeo.

[0007] En el artículo "Compresión de imágenes compuestas para la transmisión de imágenes en pantalla de ordenador en tiempo real" de T. Lin y Pengwei Hao, 2005 (Trans. Img. Proc. 14, 8 (agosto de 2005), 993 a 1005), se presenta un algoritmo de compresión de imagen compuesta que primero segmenta una imagen compuesta en píxeles de texto/gráficos y píxeles pictóricos, y luego comprime los píxeles de texto/gráficos con un nuevo algoritmo de codificación sin pérdidas y los píxeles pictóricos con el JPEG con pérdida estándar, respectivamente. La segmentación primero clasifica los bloques de imagen en bloques de imagen y de texto/gráficos mediante el uso de umbral en la cantidad de colores de cada bloque, luego extrae primitivas de forma de texto/gráficos de los bloques de imagen. La paleta de colores dinámica que rastrea los colores de texto/gráficos recientes se usa para separar primitivas de formas pequeñas del texto/gráficos a partir de los píxeles pictóricos. Las primitivas de formas también se extraen de bloques de texto/gráficos. Todas las primitivas de forma de ambos tipos de bloques se comprimen sin pérdida utilizando un algoritmo combinado de codificación basado en formas y paletas. Luego, el flujo de bits codificado sin pérdida se suministra a un codificador LZW.

[0008] En el documento N.º JCTVC-E145 "Herramientas de intracodificación e intercodificación para contenidos de pantalla" de C. Lan et al (Equipo colaborativo conjunto sobre codificación de vídeo (JCT-VC) de UIT-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11 - 5ª conferencia: Ginebra, CH, 16 a 23 de marzo de 2011), se presenta una herramienta LZ sin pérdidas basada en LZMA para mejorar la codificación de la imagen de pantalla. Las dos herramientas anteriores para la intracodificación, la cuantización escalar residual y los colores base y el mapa de índices, se extienden a la codificación entre tramas y las herramientas correspondientes se desarrollan en consecuencia para mejorar la eficacia de codificación de los vídeos de pantalla.

SUMARIO

[0009] Las técnicas de esta divulgación se refieren a la codificación de vídeo basada en paletas. Por ejemplo, en la codificación basada en paletas, un codificador de vídeo (un codificador de vídeo o un decodificador de vídeo) puede formar una denominada "paleta" como una tabla de colores para representar los datos de vídeo del área particular (por ejemplo, un bloque dado). La codificación basada en paletas puede ser especialmente útil para codificar áreas de datos de vídeo que tienen un número relativamente pequeño de colores. En lugar de codificar valores de píxeles reales (o sus residuos), el codificador de vídeo puede codificar valores de índices para uno o más de los píxeles que relacionan los píxeles con entradas en la paleta que representa los colores de los píxeles. Una paleta puede codificarse explícitamente y enviarse al decodificador, predecirse a partir de entradas de paleta anteriores, o una combinación de ambos casos. Las técnicas descritas en esta divulgación también pueden incluir técnicas para diversas combinaciones de una o más entre la señalización de modalidades de codificación basadas en paletas, la transmisión de paletas, la predicción de paletas, la obtención de paletas y la codificación de mapas de codificación basados en paletas y otros elementos sintácticos.

[0010] En un ejemplo, un procedimiento para decodificar datos de vídeo incluye determinar una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxel, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en entradas de paleta asociadas a paletas de una pluralidad de bloques de datos de vídeo; determinar, para una segunda paleta asociada a un segundo bloque de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y decodificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la decodificación de los píxeles comprende: obtener, a partir de un flujo de bits codificado, respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índices identifican una entrada de la segunda paleta; y determinar valores para los uno o más píxeles, apareando los respectivos valores de índices con al menos una de las entradas de la segunda paleta.

[0011] Preferentemente, el procedimiento comprende además determinar una frecuencia con la que cada entrada de predictor de paleta de la paleta predictora se usa para codificar píxeles de la pluralidad de bloques; y decodificar datos que identifican cada entrada de predictor de paleta de la paleta predictora, basándose en la frecuencia.

[0012] En otro ejemplo, un procedimiento para codificar datos de vídeo comprende determinar una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxel, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en entradas de paleta asociadas a paletas de una pluralidad de bloques de datos de vídeo; determinar, para una segunda paleta asociada a un segundo bloque de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta de la segunda paleta, basándose

en las entradas de predictor de paleta de la paleta predictora; y codificar los píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la codificación de los píxeles comprende: determinar los respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índices identifican las entradas respectivas de la segunda paleta; y codificar los valores de índices en un flujo de bits codificado.

[0013] En otro ejemplo, un medio no transitorio legible por ordenador almacena en el mismo instrucciones que, cuando se ejecutan, hacen que uno o más procesadores lleven a cabo uno de los procedimientos anteriormente descritos.

[0014] En otro ejemplo, un aparato para decodificar datos de vídeo incluye medios para determinar una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxel, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en entradas de paleta asociadas a una o más paletas de una pluralidad de bloques de datos de vídeo; medios para determinar, para una segunda paleta asociada a un segundo bloque de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y medios para decodificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la decodificación de los píxeles comprende: obtener, a partir de un flujo de bits codificado, respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índices identifican una entrada de la segunda paleta; y determinar los valores para los uno o más píxeles, apareando los respectivos valores de índices con al menos una de las entradas de la segunda paleta.

[0015] En otro ejemplo, se proporciona un aparato para codificar datos de vídeo, que comprende: medios para determinar una paleta de predictor que tiene una pluralidad de entradas de predictor de paleta que indican valores de píxel respectivos, en donde determinar la paleta de predictor comprende determinar la paleta de predictor basándose en la paleta entradas asociadas con una o más paletas de una pluralidad de bloques de datos de vídeo; medios para determinar, para una segunda paleta asociada con un segundo bloque de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta de la segunda paleta basadas en las entradas de predictor de paleta de la paleta de predictor; y medios para codificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la codificación de los píxeles comprende: determinar valores de índice respectivos para uno o más de los píxeles, en donde los valores de índice respectivos identifican entradas respectivas de la segunda paleta; y codificar los valores de índice en un flujo de bits codificado.

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

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0017]

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

La figura 2 es un diagrama de bloques que ilustra un codificador de vídeo ejemplar que puede implementar las técnicas descritas en esta divulgación.

La figura 3 es un diagrama de bloques que ilustra un decodificador de vídeo ejemplar que puede implementar las técnicas descritas en esta divulgación.

La figura 4 es un diagrama conceptual que ilustra un ejemplo de determinación de una paleta para codificar datos de vídeo, congruente con las técnicas de esta divulgación.

La figura 5 es un diagrama conceptual que ilustra un ejemplo de determinación de índices para una paleta para un bloque de píxeles, congruente con las técnicas de esta divulgación.

La figura 6 es un diagrama de flujo que ilustra un proceso ejemplar para codificar datos de vídeo usando una modalidad de codificación de paleta, congruente con las técnicas de esta divulgación.

La figura 7 es un diagrama de flujo que ilustra un proceso ejemplar para determinar una paleta en la codificación basada en paletas, congruente con las técnicas de esta divulgación.

La figura 8 es un diagrama de flujo que ilustra un proceso ejemplar para determinar índices de un bloque de datos de vídeo en la codificación de vídeo basada en paletas, congruente con las técnicas de esta divulgación.

5 DESCRIPCIÓN DETALLADA

[0018] Esta divulgación incluye técnicas para codificación y compresión de vídeo. En general, esta divulgación describe técnicas para la codificación basada en paletas de datos de vídeo. En la codificación de vídeo tradicional, se supone que las imágenes son de tono continuo y espacialmente fluidas. Basándose en estas suposiciones, se han desarrollado diversas herramientas, tales como la transformación basada en bloques, el filtrado, etc., y dichas herramientas han demostrado un buen rendimiento para vídeos de contenido natural.

[0019] Sin embargo, en aplicaciones como escritorio remoto, trabajo colaborativo y pantalla inalámbrica, el contenido de pantalla generado por ordenador (por ejemplo, tal como texto o gráficos de ordenador) puede ser el contenido dominante a comprimir. Este tipo de contenido tiende a tener tonos discretos, y presenta líneas nítidas y fronteras de objetos de alto contraste. Es posible que la suposición de tono continuo y suavidad ya no se aplique al contenido de la pantalla, y por tanto que las técnicas tradicionales de codificación de vídeo puedan no ser formas eficientes de comprimir los datos de vídeo, incluido el contenido de la pantalla.

[0020] Esta divulgación describe la codificación basada en paletas, que puede ser particularmente adecuada para la codificación de contenido generado por la pantalla. Por ejemplo, suponiendo que un área particular de datos de vídeo tiene un número relativamente pequeño de colores. Un codificador de vídeo (un codificador de vídeo o un decodificador de vídeo) puede formar una denominada "paleta" como una tabla de colores para representar los datos de vídeo del área particular (por ejemplo, un bloque dado). Cada píxel puede asociarse a una entrada en la paleta que representa el color del píxel. Por ejemplo, el codificador de vídeo puede codificar un índice que relaciona el valor de píxel con el valor adecuado en la paleta.

[0021] En el ejemplo anterior, un codificador de vídeo puede codificar un bloque de datos de vídeo determinando una paleta para el bloque (por ejemplo, codificando la paleta explícitamente, prediciéndola o combinando ambos casos), ubicando una entrada en la paleta para representar el valor de cada píxel y codificando el bloque con valores de índices para los píxeles que relacionan el valor de píxel con la paleta. Un decodificador de vídeo puede obtener, a partir de un flujo de bits codificado, una paleta para un bloque, así como valores de índices para los píxeles del bloque. El decodificador de vídeo puede relacionar los valores de índice de los píxeles con las entradas de la paleta para reconstruir los valores de píxel del bloque.

[0022] El ejemplo anterior pretende proporcionar una descripción general de la codificación basada en paletas. En diversos ejemplos, las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de una o más entre la señalización de modalidades de codificación basadas en paletas, la transmisión de paletas, la predicción de paletas, la obtención de paletas y la transmisión de mapas de codificación basados en paletas y otros elementos sintácticos. Dichas técnicas pueden mejorar la eficiencia de la codificación de vídeo, por ejemplo, requerir menos bits para representar el contenido generado por la pantalla.

[0023] Las técnicas para la codificación basada en paletas de datos de vídeo pueden usarse con otras una o más técnicas de codificación diferentes, tales como técnicas para la codificación interpredictiva o intrapredictiva. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o decodificador, o un codificador-decodificador combinado (códec), se puede configurar para realizar una codificación interpredictiva e intrapredictiva, así como una codificación basada en paletas.

[0024] En algunos ejemplos, las técnicas de codificación basadas en paletas pueden configurarse para su uso con una o más normas de codificación de vídeo. La Codificación de Vídeo de Alta Eficacia (HEVC) es una nueva norma de codificación de vídeo desarrollada por el Equipo Conjunto de Colaboración en Codificación de vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo del UIT-T (VCEG) y el Grupo de Expertos en Imagen en Movimiento de ISO/IEC (MPEG). Un reciente borrador de especificación textual de la HEVC se describe en el artículo de Bross et al., "High Efficiency video Coding (HEVC) Text Specification Draft 10 (for FDIS & Consent)" ["Borrador de especificación textual 10 de la codificación de vídeo de alta eficacia (HEVC) (para FDIS y consentimiento)"], JCVCL1003_v13, 12ª conferencia del JCT-VC de ITU-T SG16 WP 3 e ISO/IEC JCT 1/SC 29/WG 11, 14 al 23 de enero de 2013 ("Borrador 10 de HEVC"), disponible en: http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v13.zip.

[0025] Con respecto al marco de la HEVC, como ejemplo, las técnicas de codificación basada en paletas se pueden configurar para usarse como un modo de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basada en paletas pueden configurarse para usarse como una modalidad de PU en el entorno de la HEVC. Por consiguiente, todos los procesos divulgados a continuación, descritos en el contexto de un modo CU, pueden, adicional o alternativamente, aplicarse a PU. Sin embargo, estos ejemplos basados en la HEVC no deberían considerarse como una restricción o limitación de las técnicas de codificación basada en paletas, descritas en el presente documento, ya que dichas técnicas se pueden aplicar para que funcionen independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación por paleta pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[0026] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 ejemplar que puede utilizar las técnicas de esta divulgación. Como se usa en el presente documento, el término "codificador de vídeo" se refiere genéricamente tanto a codificadores de vídeo como a decodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o a la descodificación de vídeo. El codificador de vídeo 20 y el decodificador de vídeo 30 del sistema de codificación de vídeo 10 representan ejemplos de dispositivos que pueden configurarse para realizar técnicas de codificación de vídeo basada en paletas de acuerdo con diversos ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar configurados para codificar selectivamente varios bloques de datos de vídeo, tales como las CU o PU en la codificación HEVC, usando la codificación basada en paletas o bien la codificación no basada en paletas. Las modalidades de codificación no basada en paletas pueden referirse a diversas modalidades de codificación temporal interpredictiva o a modalidades de codificación espacial intrapredictiva, tales como las diversas modalidades de codificación especificadas por el Borrador 10 de la HEVC.

[0027] Como se muestra en la FIG. 1, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12 y un dispositivo de destino 14. El dispositivo de origen 12 genera datos de vídeo codificados. Por consiguiente, el dispositivo de origen 12 puede denominarse dispositivo de codificación de vídeo o aparato de codificación de vídeo. El dispositivo de destino 14 puede decodificar datos de vídeo codificados, generados por el dispositivo de origen 12. Por consiguiente, el dispositivo de destino 14 puede denominarse dispositivo de descodificación de vídeo o aparato de descodificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo.

[0028] El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, dispositivos informáticos móviles, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos telefónicos manuales tales como los denominados teléfonos "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, ordenadores de coche o similares.

[0029] El dispositivo de destino 14 puede recibir datos de vídeo codificados desde el dispositivo de origen 12 a través de un canal 16. El canal 16 puede comprender uno o más medios o dispositivos capaces de mover los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el canal 16 puede comprender uno o más medios de comunicación que permitan al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. En este ejemplo, el dispositivo de origen 12 puede modular los datos de vídeo codificados de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y puede transmitir los datos de vídeo modulados al dispositivo de destino 14. Los uno o más medios de comunicación pueden incluir medios de comunicación inalámbricos y/o alámbricos, tales como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. Los uno o más medios de comunicación pueden formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global (por ejemplo, Internet). Los uno o más medios de comunicación pueden incluir routers, conmutadores, estaciones base u otros equipos que faciliten la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0030] En otro ejemplo, el canal 16 puede incluir un medio de almacenamiento que almacene datos de vídeo codificados, generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder al medio de almacenamiento a través del acceso al disco o del acceso a la tarjeta. El medio de almacenamiento puede incluir varios medios de almacenamiento de datos de acceso local, tales como discos Blu-ray, DVD, CD-ROM, memoria flash u otros medios adecuados de almacenamiento digital para almacenar datos de vídeo codificados.

[0031] En un ejemplo adicional, el canal 16 puede incluir un servidor de ficheros u otro dispositivo de almacenamiento intermedio que almacene los datos de vídeo codificados, generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder a datos de vídeo codificados, almacenados en el servidor de ficheros o en otro dispositivo de almacenamiento intermedio mediante transmisión continua o descarga. El servidor de archivos puede ser un tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir los datos de vídeo codificados al dispositivo de destino 14. Entre los ejemplos de servidores de archivos se incluyen servidores de red (por ejemplo, para una página web), servidores del protocolo de transferencia de archivos (FTP), dispositivos de almacenamiento conectados a red (NAS) y unidades de disco local.

[0032] El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de una conexión de datos estándar, tal como una conexión a Internet. Entre los ejemplos de tipos de conexiones de datos pueden incluirse canales inalámbricos (por ejemplo, conexiones WiFi), conexiones cableadas (por ejemplo, DSL, módem de cable, etc.) o combinaciones de ambos que sean adecuadas para acceder a datos de vídeo codificados almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificados desde el servidor de archivos puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0033] Las técnicas de esta divulgación no están limitadas a aplicaciones o a configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo como soporte de varias aplicaciones de multimedia, tales como difusiones de televisión por el aire, radiodifusiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo, por ejemplo, mediante Internet, codificación de datos de vídeo para su almacenamiento en un medio de almacenamiento de datos, decodificación de datos de vídeo almacenados en un medio de almacenamiento de datos u otras aplicaciones. En algunos ejemplos, el sistema de codificación de vídeo 10 puede configurarse para soportar la transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

[0034] La FIG. 1 es meramente un ejemplo y las técnicas de esta divulgación pueden aplicarse a configuraciones de codificación de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no incluyen necesariamente alguna comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, los datos se recuperan desde una memoria local, se transmiten continuamente por una red o similar. Un dispositivo de codificación de vídeo puede codificar y almacenar datos en la memoria, y/o un dispositivo de decodificación de vídeo puede recuperar y decodificar datos desde la memoria. En muchos ejemplos, la codificación y la decodificación se realiza mediante dispositivos que no se comunican entre sí, sino que simplemente codifican datos en la memoria y/o recuperan y decodifican datos desde la memoria.

[0035] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye un origen de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos ejemplos, la interfaz de salida 22 puede incluir un modulador/demodulador (módem) y/o un transmisor. La fuente de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contenga datos de vídeo previamente capturados, una interfaz de alimentación de vídeo para recibir datos de vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de vídeo, o una combinación de dichas fuentes de datos de vídeo.

[0036] El codificador de vídeo 20 puede codificar datos de vídeo procedentes de la fuente de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 transmite directamente los datos de vídeo codificados al dispositivo de destino 14 mediante la interfaz de salida 22. En otros ejemplos, los datos de vídeo codificados también pueden almacenarse en un medio de almacenamiento o en un servidor de archivos para un acceso posterior mediante el dispositivo de destino 14 para su decodificación y/o su reproducción.

[0037] En el ejemplo de la FIG. 1, el dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos ejemplos, la interfaz de entrada 28 incluye un receptor y/o un módem. La interfaz de entrada 28 puede recibir los datos de vídeo codificados por el canal 16. El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En general, el dispositivo de visualización 32 muestra los datos de vídeo decodificados. El dispositivo de visualización 32 puede comprender varios dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[0038] Esta divulgación puede referirse, en general, al codificador de vídeo 20 "que señala" o "que transmite" cierta información a otro dispositivo, tal como el decodificador de vídeo 30. El término "señalar" o "transmitir" puede referirse

en general a la comunicación de elementos sintácticos y/u otros datos usados para decodificar los datos de vídeo comprimidos. Dicha comunicación puede producirse en tiempo real o casi real. De forma alternativa, dicha comunicación puede producirse durante un tramo de tiempo, tal como podría producirse cuando se almacenan elementos sintácticos en un medio de almacenamiento legible por ordenador en un flujo de bits codificado en el momento de la codificación, que entonces pueden ser recuperado por un dispositivo de descodificación en cualquier momento tras haber sido almacenado en este medio. Por lo tanto, aunque el decodificador de vídeo 30 puede mencionarse como "recibiendo"/ "que recibe" cierta información, la recepción de información no necesariamente se produce en tiempo real o casi real, y puede recuperarse desde un medio en algún momento después del almacenamiento.

[0039] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden, cada uno, implementarse como cualquiera entre varios circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA), lógica discreta, hardware o cualquier combinación de los mismos. Si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado de almacenamiento no transitorio legible por ordenador, y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cualquiera de los anteriores (incluyendo hardware, software, una combinación de hardware y software, etc.) puede considerarse como uno o más procesadores. Cada uno entre el codificador de vídeo 20 y el decodificador de vídeo 30 se puede incluir en uno o más codificadores o descodificadores, cualquiera de los cuales se puede integrar como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo.

[0040] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 funcionan de acuerdo con una norma de compresión de vídeo, tal como la norma HEVC mencionada anteriormente y descrita en el Borrador 10 de la HEVC. Además de la norma base de HEVC, hay esfuerzos constantes para producir extensiones de codificación de vídeo ajustable a escala, codificación de vídeo multivistas y codificación 3D para la HEVC. Además, pueden proporcionarse modalidades de codificación basada en paletas, por ejemplo, como se describe en esta divulgación, para la extensión de la norma HEVC. En algunos ejemplos, las técnicas descritas en esta divulgación para la codificación basada en paletas se pueden aplicar a codificadores y decodificadores configurados para funcionar de acuerdo con otras normas de codificación de vídeo, tales como la norma ITU-T H.264/AVC o normas futuras. Por consiguiente, la aplicación de una modalidad de codificación basada en paletas para la codificación de unidades de codificación (CU) o unidades de predicción (PU) en un códec de HEVC se describe con fines de ejemplo.

[0041] En la HEVC y en otras normas de codificación de vídeo, una secuencia de vídeo incluye típicamente una serie de imágenes. Las imágenes también pueden denominarse "tramas". Una imagen puede incluir tres matrices de muestras, indicadas con S_L , S_{Cb} y S_{Cr} . S_L es una matriz bidimensional (es decir, un bloque) de muestras de luma. S_{Cb} es una matriz bidimensional de muestras de croma Cb. S_{Cr} es una matriz bidimensional de muestras de croma Cr. Las muestras de crominancia también se pueden denominar en el presente documento muestras de "croma". En otros casos, una imagen puede ser monocromática y puede solo incluir una matriz de muestras de luminancia.

[0042] Para generar una representación codificada de una imagen, el codificador de vídeo 20 puede generar un conjunto de unidades arboladas de codificación (CTU). Cada una de las CTU puede ser un bloque arbolado de codificación de muestras de luminancia, dos correspondientes bloques arbolados de codificación de muestras de crominancia y estructuras sintácticas usadas para codificar las muestras de los bloques arbolados de codificación. Un bloque de árbol de codificación puede ser un bloque de muestras de tamaño $N \times N$. Una CTU también puede denominarse "bloque arbolado" o "máxima unidad de codificación" (LCU). Las CTU de HEVC pueden ser análogas en gran medida a los macrobloques de otras normas, tales como H.264/AVC. Sin embargo, una CTU no está limitada necesariamente a un tamaño particular y puede incluir una o más unidades de codificación (CU). Un fragmento puede incluir un número entero de CTU ordenadas consecutivamente en el escaneo de cuadrícula.

[0043] Un fragmento codificado puede comprender una cabecera de fragmento y datos de fragmento. La cabecera de fragmento de un fragmento puede ser una estructura sintáctica que incluye elementos sintácticos que proporcionan información sobre el fragmento. Los datos del fragmento pueden incluir CTU codificadas del fragmento.

[0044] Esta divulgación puede usar el término "unidad de vídeo", "bloque de vídeo" o "bloque" para referirse a uno o más bloques de muestras y estructuras sintácticas usadas para codificar muestras del uno o más bloques de muestras. Los tipos de ejemplo de unidades de vídeo pueden incluir CTU, CU, PU, unidades de transformadas (TU), macrobloques, particiones de macrobloque, etc. En algunos contextos, el análisis de las PU puede intercambiarse con el análisis de los macrobloques o particiones de macrobloques.

[0045] Para generar una CTU codificada, el codificador de vídeo 20 puede realizar de forma recursiva una partición de árbol cuaternario en los bloques de árbol de codificación de una CTU para dividir los bloques de árbol de codificación en bloques de codificación, de ahí el nombre "unidades de árbol de codificación". Un bloque de codificación es un bloque de muestras de tamaño NxN. Una CU puede ser un bloque de codificación de muestras de luminancia y dos correspondientes bloques de codificación de muestras de crominancia de una imagen que tenga una formación de muestras de luminancia, una formación de muestras de Cb y una formación de muestras de Cr, y estructuras sintácticas usadas para codificar las muestras de los bloques de codificación. El codificador de vídeo 20 puede dividir un bloque de codificación de una CU en uno o más bloques de predicción. Un bloque de predicción puede ser un bloque rectangular (es decir, cuadrado o no cuadrado) de muestras en las que se aplique la misma predicción. Una unidad de predicción (PU) de una CU puede ser un bloque de predicción de muestras de luma, dos correspondientes bloques de predicción de muestras de croma de una imagen y estructuras sintácticas usadas para predecir las muestras de bloques de predicción. El codificador de vídeo 20 puede generar bloques predictivos de luma, Cb y Cr para bloques de predicción de luma, Cb y Cr de cada PU de la CU.

[0046] El codificador de vídeo 20 puede usar intrapredicción o interpredicción para generar los bloques predictivos para una PU. Si el codificador de vídeo 20 usa la intrapredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras decodificadas de la imagen asociada a la PU.

[0047] Si el codificador de vídeo 20 usa predicción inter para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras descodificadas de una o más imágenes distintas a la imagen asociada a la PU. El codificador de vídeo 20 puede usar unipredicción o bipredicción para generar los bloques predictivos de una PU. Cuando el codificador de vídeo 20 usa la unipredicción para generar los bloques predictivos para una PU, la PU puede tener un MV único. Cuando el codificador de vídeo 20 usa bipredicción para generar los bloques predictivos para una PU, la PU puede tener dos MV.

[0048] Después de que el codificador de vídeo 20 genera bloques predictivos de luma, Cb y Cr para una o más PU de una CU, el codificador de vídeo 20 puede generar un bloque residual de luma para la CU. Cada muestra en el bloque residual de luma de la CU indica una diferencia entre una muestra de luma en uno de los bloques predictivos de luma de la CU y una muestra correspondiente en el bloque de codificación de luma original de la CU. Además, el codificador de vídeo 20 puede generar un bloque residual de Cb para la CU. Cada muestra en el bloque residual de Cb de la CU puede indicar una diferencia entre una muestra de Cb en uno de los bloques predictivos de Cb de la CU y una muestra correspondiente en el bloque de codificación de Cb original de la CU. El codificador de vídeo 20 también puede generar un bloque residual de Cr para la CU. Cada muestra en el bloque residual de Cr de la CU puede indicar una diferencia entre una muestra de Cr en uno de los bloques predictivos de Cr de la CU y una muestra correspondiente en el bloque de codificación de Cr original de la CU.

[0049] Además, el codificador de vídeo 20 puede usar la partición en árbol cuádruple para descomponer los bloques residuales de luma, Cb y Cr de una CU en uno o más bloques de transformación de luma, Cb y Cr. Un bloque de transformación puede ser un bloque rectangular de muestras en el que se aplique la misma transformación. Una unidad de transformación (TU) de una CU puede ser un bloque de transformación de muestras de luma, dos bloques de transformación correspondientes de muestras de croma y estructuras sintácticas usadas para transformar las muestras de bloques de transformación. De este modo, cada TU de una CU puede estar asociada a un bloque de transformación de luma, un bloque de transformación de Cb y un bloque de transformación de Cr. El bloque de transformada de luma asociado con la TU puede ser un subbloque del bloque residual de luma de la CU. El bloque de transformada de Cb puede ser un subbloque del bloque residual de Cb de la CU. El bloque de transformada de Cr puede ser un subbloque del bloque residual de Cr de la CU.

[0050] El codificador de vídeo 20 puede aplicar una o más transformaciones a un bloque de transformación de luma de una TU con el fin de generar un bloque de coeficientes de luma para la TU. Un bloque de coeficientes puede ser una formación bidimensional de coeficientes de transformación. Un coeficiente de transformada puede ser una cantidad escalar. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cb de una TU para generar un bloque de coeficientes de Cb para la TU. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cr de una TU para generar un bloque de coeficientes de Cr para la TU.

[0051] Después de generar un bloque de coeficientes (por ejemplo, un bloque de coeficientes de luma, un bloque de coeficientes de Cb o un bloque de coeficientes de Cr), el codificador de vídeo 20 puede cuantizar el bloque de coeficientes.

La cuantización se refiere, en general, a un proceso en el que los coeficientes de transformación se cuantizan para reducir, posiblemente, la cantidad de datos usados para representar los coeficientes de transformación, proporcionando una compresión adicional. Después de que el codificador de vídeo 20 cuantice un bloque de coeficientes, el codificador de vídeo 20 puede codificar por entropía elementos sintácticos que indiquen los coeficientes de transformada cuantizados. Por ejemplo, el codificador de vídeo 20 puede realizar la Codificación Aritmética Binaria Adaptativa al Contexto (CABAC) en los elementos sintácticos que indiquen los coeficientes de transformada cuantificados. El codificador de vídeo 20 puede emitir los elementos sintácticos codificados por entropía en un flujo de bits.

[0052] El codificador de vídeo 20 puede emitir un flujo de bits que incluya los elementos sintácticos codificados por entropía. El flujo de bits puede incluir una secuencia de bits que forme una representación de imágenes codificadas y datos asociados. El flujo de bits puede comprender una secuencia de unidades de capas de abstracción de red (NAL). Cada una de las unidades de NAL incluye un encabezado de unidad de NAL y encapsula una carga útil de secuencias de octetos sin procesar (RBSP). La cabecera de la unidad de NAL puede incluir un elemento sintáctico que indique un código de tipo de unidad de NAL. El código de tipo de unidad NAL especificado por la cabecera de unidad NAL de una unidad NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contenga un número entero de bytes que se encapsule dentro de una unidad de NAL. En algunos casos, una RBSP incluye cero bits.

[0053] Diferentes tipos de unidades NAL pueden encapsular diferentes tipos de RBSP. Por ejemplo, un primer tipo de unidad de NAL puede encapsular una RBSP para un conjunto de parámetros de imagen (PPS), un segundo tipo de unidad de NAL puede encapsular una RBSP para un fragmento codificado, un tercer tipo de unidad de NAL puede encapsular una RBSP para SEI, etc. Las unidades de NAL que encapsulan las RBSP para datos de codificación de vídeo (a diferencia de las RBSP para conjuntos de parámetros y mensajes de SEI) pueden denominarse unidades de NAL de la capa de codificación de vídeo (VCL).

[0054] El decodificador de vídeo 30 puede recibir un flujo de bits generado por el codificador de vídeo 20. Además, el decodificador de vídeo 30 puede analizar sintácticamente el flujo de bits para decodificar elementos sintácticos del flujo de bits. El decodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo basándose, al menos en parte, en los elementos sintácticos decodificados a partir del flujo de bits. El proceso para reconstruir los datos de vídeo puede ser, en general, recíproco al proceso realizado por el codificador de vídeo 20.

[0055] Por ejemplo, el decodificador de vídeo 30 puede usar los MV de las PU para determinar bloques predictivos de muestra para las PU de una CU actual. Además, el decodificador de vídeo 30 puede cuantizar inversamente bloques de coeficientes de transformación, asociados a las TU de la CU actual. El decodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes de transformada para reconstruir los bloques de transformada asociados a las TU de la CU actual. El decodificador de vídeo 30 puede reconstruir los bloques de codificación de la CU actual añadiendo las muestras de los bloques predictivos de muestra para las PU de la CU actual a las muestras correspondientes de los bloques de transformación de las TU de la CU actual. Mediante la reconstrucción de los bloques de codificación para cada CU de una imagen, el decodificador de vídeo 30 puede reconstruir la imagen.

[0056] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para realizar una codificación basada en paletas. Por ejemplo, en la codificación basada en paletas, en lugar de realizar las técnicas de codificación intrapredictiva o interpredictiva, descritas anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar una denominada paleta como una tabla de colores para representar los datos de vídeo de un área particular (por ejemplo, un bloque dado). Cada píxel puede asociarse a una entrada en la paleta que representa el color del píxel. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar un índice que relaciona el valor de píxel con el valor adecuado en la paleta.

[0057] En el ejemplo anterior, el codificador de vídeo 20 puede codificar un bloque de datos de vídeo determinando una paleta para el bloque, ubicando una entrada en la paleta para representar el valor de cada píxel y codificando la paleta con valores de índice para los píxeles relacionando el valor de píxel con la paleta. El decodificador de vídeo 30 puede obtener, a partir de un flujo de bits codificado, una paleta para un bloque, así como valores de índice para los píxeles del bloque. El decodificador de vídeo 30 puede relacionar los valores de índice de los píxeles con las entradas de paleta para reconstruir los valores de píxel del bloque.

[0058] La codificación basada en paleta puede tener una cierta cantidad de sobrecarga de señalización. Por ejemplo, se pueden necesitar varios bits para señalar las características de una paleta, tales como el tamaño de la paleta, así como la paleta misma. Además, se pueden necesitar varios bits para indicar valores de índice para los píxeles del bloque.

Las técnicas de esta divulgación pueden, en algunos ejemplos, reducir el número de bits necesarios para señalar dicha información. Las técnicas descritas en esta divulgación también pueden incluir técnicas para diversas combinaciones de una o más entre la señalización de modalidades de codificación basados en paletas, la transmisión de paletas, la predicción de paletas, la obtención de paletas y la transmisión de mapas de codificación basados en paletas y otros elementos sintácticos.

[0059] Los aspectos de esta divulgación están dirigidos a la predicción de la paleta. Por ejemplo, según los aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar una primera paleta que tenga primeras entradas que indiquen primeros valores de píxeles. El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden entonces determinar, basándose en las primeras entradas de la primera paleta, una o más segundas entradas que indican los segundos valores de píxeles de una segunda paleta. El codificador de vídeo 20 y/o el decodificador de vídeo 30 también pueden codificar píxeles de un bloque de datos de vídeo usando la segunda paleta.

[0060] Cuando se determinan las segundas entradas de la segunda paleta basándose en las primeras entradas, el codificador de vídeo 20 puede codificar varios elementos sintácticos, que pueden ser utilizados por el decodificador de vídeo para reconstruir la segunda paleta. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos en un flujo de bits para indicar que una paleta completa (o paletas, en el caso de cada componente cromático, por ejemplo, Y, Cb, Cr o Y, U, V o R, G, B, de los datos de vídeo que tienen una paleta independiente) se copia desde uno o más bloques vecinos del bloque que se está codificando actualmente. La paleta a partir de la cual se pronostican las entradas de la paleta actual del bloque actual (por ejemplo, copiada) puede denominarse una paleta predictiva. La paleta predictiva puede contener entradas de paleta procedentes de uno o más bloques vecinos, incluidos los bloques espacialmente vecinos y/o los bloques vecinos en un orden de exploración particular de los bloques. Por ejemplo, los bloques vecinos pueden estar ubicados espacialmente a la izquierda (bloque vecino izquierdo) o por encima (bloque vecino superior) del bloque que se está codificando actualmente. En otro ejemplo, el codificador de vídeo 20 puede determinar entradas de la paleta predictiva utilizando los valores de muestra más frecuentes en un vecino casual del bloque actual. En otro ejemplo, los bloques vecinos pueden colindar con el bloque que se está codificando actualmente, de acuerdo a un orden de exploración particular utilizado para codificar los bloques. Es decir, los bloques vecinos pueden ser uno o más bloques codificados antes del bloque actual en el orden de exploración. El codificador de vídeo 20 puede codificar uno o más elementos sintácticos para indicar la ubicación de los bloques vecinos desde los cuales se copian las una o más paletas.

[0061] En algunos ejemplos, la predicción de paleta se puede realizar para cada entrada. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para indicar, para cada entrada de una paleta predictiva, si la entrada de la paleta está incluida en la paleta para el bloque actual. Si el codificador de vídeo 20 no predice una entrada de la paleta para el bloque actual, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos adicionales para especificar las entradas no predichas, así como el número de dichas entradas.

[0062] En algunos ejemplos, las técnicas para predecir una paleta completa se pueden combinar con técnicas para predecir una o más entradas de una paleta. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos en un flujo de bits para indicar si la paleta actual está copiada completamente a partir de la paleta predictiva. Si este no es el caso, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos en un flujo de bits para indicar si cada entrada en la paleta predictiva está copiada.

[0063] En otro ejemplo, en lugar de señalar el número de entradas y el valor de paleta, el codificador de vídeo 20 puede señalar, después de señalar cada valor de paleta, un indicador para indicar si el valor de paleta señalado es la entrada final de paleta para la paleta. El codificador de vídeo 20 puede no señalar un indicador tal de "fin de paleta" si la paleta ya ha alcanzado un cierto tamaño máximo.

[0064] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para indicar si la predicción de paleta está habilitada y/o activa. En un ejemplo con fines de ilustración, el codificador de vídeo 20 puede codificar un indicador _paleta_pred para indicar, para cada bloque (por ejemplo, una CU o una PU), si el codificador de vídeo 20 usa la predicción por paletas para predecir la paleta para el bloque respectivo. En algunos ejemplos, el codificador de vídeo puede señalar un indicador independiente para cada componente cromático (por ejemplo, tres indicadores para cada bloque). En otros ejemplos, el codificador de vídeo 20 puede señalar un único indicador que es aplicable a todos los componentes cromáticos de un bloque.

[0065] El decodificador de vídeo 30 puede obtener la información identificada anteriormente a partir de un flujo de bits codificado y utilizar los datos para reconstruir la paleta. Por ejemplo, el decodificador de vídeo 30 puede recibir datos que indiquen si una paleta en particular se predice desde otra paleta, así como información que permite al decodificador de vídeo 30 usar las entradas adecuadas de paleta predictiva.

[0066] En algunos casos, adicional o alternativamente, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden construir una paleta sobre la marcha, es decir, dinámicamente. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden añadir entradas a una paleta vacía durante la codificación. Es decir, el codificador de vídeo 20 puede añadir valores de píxel a una paleta a medida que los valores de píxel se generan y transmiten para posiciones en un bloque. Los píxeles codificados relativamente más adelante en el bloque pueden referirse a entradas agregadas anteriormente de la paleta, por ejemplo, con valores de índices, en lugar de transmitir los valores de píxeles. Del mismo modo, al recibir un nuevo valor de píxel para una posición en un bloque, el decodificador de vídeo 30 puede seguir el mismo proceso que el codificador de vídeo 20 e incluir el valor de píxel en una paleta. De esta manera, el decodificador de vídeo 30 construye la misma paleta que el codificador de vídeo 20. El decodificador de vídeo 30 puede recibir, para los píxeles que tienen valores que ya están incluidos en la paleta, valores de índices que identifican los valores. El decodificador de vídeo 30 puede usar la información recibida, por ejemplo, los valores de píxeles para la paleta y los valores de índices, para reconstruir los píxeles de un bloque.

[0067] En algunos casos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden mantener una paleta de un tamaño fijo. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden añadir los valores de píxel reconstruidos más recientes a la paleta según se reconstruyen. Para cada entrada que se añade a la paleta, se descarta la entrada que se añadió a la paleta más temprano. Esto también a veces se menciona como 'primero en entrar, primero en salir' (FIFO). Este proceso de actualización de la paleta puede aplicarse solo a los bloques que se codifican utilizando la modalidad de paleta o a todos los bloques, independientemente de la modalidad de codificación.

[0068] Las técnicas descritas anteriormente se refieren en general al codificador de vídeo 20 y al decodificador de vídeo 30 que construyen y/o transmiten una paleta para la codificación basada en paleta. Otros aspectos de esta divulgación se refieren a la construcción y/o transmisión de una correlación que permite que el codificador de vídeo 20 y/o el decodificador de vídeo 30 determinen los valores de píxeles. Por ejemplo, otros aspectos de esta divulgación se refieren a la construcción y/o la transmisión de una correlación de índices que relacionan un píxel particular con una entrada de una paleta.

[0069] En algunos ejemplos, el codificador de vídeo 20 puede indicar si los píxeles de un bloque tienen un valor correspondiente en una paleta. En un ejemplo con fines de ilustración, supóngase que una entrada (i, j) de una correlación corresponde a una posición de píxel (i, j) en un bloque de datos de vídeo. En este ejemplo, el codificador de vídeo 20 puede codificar un indicador para cada posición de píxel de un bloque. El codificador de vídeo 20 puede fijar el indicador igual a uno para la entrada (i, j), para indicar que el valor de píxel en la ubicación (i, j) es uno de los valores en la paleta. Cuando se incluye un color en la paleta (es decir, el indicador es igual a uno), el codificador de vídeo 20 también puede codificar datos que indiquen un índice de paleta para la entrada (i, j) que identifica el color en la paleta. Cuando el color del píxel no se incluye en la paleta (es decir, el indicador es igual a cero), el codificador de vídeo 20 también puede codificar datos que indican un valor de muestra para el píxel. El decodificador de vídeo 30 puede obtener los datos descritos anteriormente a partir de un flujo de bits codificado y utilizar los datos para determinar un índice de paleta y/o valor de píxel para una ubicación particular en un bloque.

[0070] En algunos casos, puede haber una correlación entre el índice de paleta con el que se correlaciona un píxel en una posición dada y la probabilidad de que un píxel vecino sea correlacionado con el mismo índice de paleta. Es decir, cuando un píxel se correlaciona con un índice de paleta particular, puede ser relativamente alta la probabilidad de que uno o más píxeles vecinos (en términos de ubicación espacial) se correlacionen con el mismo índice de paleta.

[0071] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar y codificar uno o más índices de un bloque de datos de vídeo, en relación con uno o más índices del mismo bloque de datos de vídeo. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden configurarse para determinar un primer valor de índice asociado a un primer píxel en un bloque de datos de vídeo, donde el primer valor de índice relaciona un valor del primer píxel con una entrada de una paleta. El codificador de vídeo 20 y/o el decodificador de vídeo 30 también pueden configurarse para determinar, basándose en el primer valor de índice, uno o más segundos valores de índices asociados a uno o más segundos píxeles en el bloque de datos de vídeo, y para codificar los primeros y los uno o más segundos píxeles del bloque de datos de vídeo. Por lo tanto, en este ejemplo, los índices de una correlación pueden codificarse en relación con otros uno o más índices de la correlación.

[0072] En algunos ejemplos, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican una serie de píxeles consecutivos, en un orden de exploración dado, que están correlacionados con el mismo valor de índice. La cadena de valores de índice de valor similar se puede denominar en el presente documento una "serie". En algunos ejemplos, un valor de píxel puede estar asociado con exactamente un valor de índice en una paleta. En consecuencia, en algunos casos, una serie de valores también puede referirse a una cadena de valores de píxel de valor similar. En otros ejemplos, como se describe con respecto a la codificación con pérdidas a continuación, más de un valor de píxel puede correlacionarse con el mismo valor de índice en una paleta. En tales ejemplos, una racha de valores se refiere a valores de índice con iguales valores.

[0073] En un ejemplo con fines de ilustración, si dos índices consecutivos en un orden de exploración dado tienen valores diferentes, la racha es igual a cero. Si dos índices consecutivos en un orden de exploración dado tienen el mismo valor pero el tercer índice en el orden de exploración tiene un valor diferente, la racha es igual a uno. El decodificador de vídeo 30 puede obtener los elementos sintácticos que indican una racha desde un flujo de bits codificado y utilizar los datos para determinar el número de ubicaciones de píxeles consecutivas que tienen el mismo valor de índice.

[0074] Adicional o alternativamente, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden realizar la copia por líneas para una o más entradas de una correlación. Las entradas también pueden denominarse "posiciones" debido a la relación entre las entradas de la correlación y las posiciones de píxeles de un bloque. La copia por líneas puede depender, en algunos ejemplos, de la dirección de exploración. Por ejemplo, el codificador de vídeo 20 puede indicar que un valor de píxel o valor de correlación de índices para una posición particular en un bloque es igual al valor de píxel o índice en una línea por encima de la posición particular (para una exploración horizontal) o en la columna a la izquierda de la posición particular (para una exploración vertical). El codificador de vídeo 20 también puede indicar, como una racha, el número de valores de píxel o índices en el orden de exploración que son iguales a los correspondientes valores de píxel o índices arriba, o en la columna a la izquierda, de la posición particular. En este ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden copiar los valores de píxel o índice de la línea adyacente especificada y del número especificado de entradas para la línea del bloque que se está codificando actualmente.

[0075] En algunos casos, la línea desde la cual se copian los valores puede estar directamente adyacente a, por ejemplo, arriba o a la izquierda de la línea de la posición que se está codificando actualmente. En otros ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden almacenar temporalmente una serie de líneas del bloque, de manera que cualesquiera entre la serie de líneas de la correlación se puedan usar como valores predictivos para una línea de la correlación que se está codificando actualmente. En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden configurarse para almacenar las cuatro filas anteriores de índices o valores de píxel, antes de codificar la fila actual de píxeles. En este ejemplo, la fila predictiva (la fila desde la cual se copian los índices o los valores de píxeles) puede indicarse en un flujo de bits con un código unario truncado u otros códigos, tales como los códigos unarios. Con respecto a un código unario truncado, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar un valor máximo para el código unario truncado basándose en un cálculo de fila máxima (por ejemplo, índice_fila-1) o un cálculo de columna máxima (por ejemplo, índice_columna-1). Además, también se puede incluir en el flujo de bits una indicación del número de posiciones de la fila predictiva que se copian. En algunos casos, si la línea o columna desde la cual se predice una posición actual pertenece a otro bloque (por ejemplo, CU o CTU), dicha predicción puede ser inhabilitada.

[0076] De acuerdo a aspectos de esta divulgación, las técnicas para codificar las denominadas rachas de entradas pueden usarse conjuntamente con las técnicas de copiado por líneas, descritas anteriormente. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos (por ejemplo, un indicador) que indica si el valor de una entrada en una correlación se obtiene a partir de una paleta o si el valor de una entrada en la correlación se obtiene a partir de una línea codificada previamente en la correlación. El codificador de vídeo 20 también puede codificar uno o más elementos sintácticos que indican un valor de índice de una paleta o la ubicación de la entrada en la línea (la fila o columna). El codificador de vídeo 20 también puede codificar uno o más elementos sintácticos que indican un número de entradas consecutivas que comparten el mismo valor. El decodificador de vídeo 30 puede obtener dicha información a partir de un flujo de bits codificado y usar la información para reconstruir la correlación y los valores de píxeles para un bloque.

[0077] Como se ha señalado anteriormente, los índices de una correlación se recorren en un orden particular. De acuerdo a aspectos de esta divulgación, la dirección de exploración puede ser vertical, horizontal o en diagonal (por

ejemplo, 45 grados o 135 grados diagonalmente en bloque). En algunos ejemplos, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para cada bloque que indica una dirección de exploración para recorrer los índices del bloque. Adicional o alternativamente, la dirección de exploración puede ser señalizada o deducida basándose en la llamada información lateral, tal como, por ejemplo, tamaño de bloque, espacio cromático y/o componente cromático.

5 El codificador de vídeo 20 puede especificar exploraciones para cada componente cromático de un bloque. Alternativamente, una exploración específica puede aplicarse a todos los componentes cromáticos de un bloque.

[0078] Las técnicas de esta divulgación también incluyen otros aspectos de la codificación basada en paletas. Por ejemplo, según los aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar uno o más elementos sintácticos para cada bloque, para indicar que el bloque está codificado usando una modalidad de codificación de paleta. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar un indicador de modalidad de paleta (indicador_Modalidad_PLT) para indicar si se ha de usar una modalidad de codificación basada en paleta para codificar un bloque en particular. En este ejemplo, el codificador de vídeo 20 puede codificar un indicador_Modalidad_PLT que es igual a uno, para especificar que el bloque que se está codificando actualmente ("bloque actual") se codifica usando una modalidad de paleta. Un valor del indicador_Modalidad_PLT igual a

10 cero especifica que el bloque actual no está codificado utilizando la modalidad de paleta. En este caso, el decodificador de vídeo 30 puede obtener el indicador_Modalidad_PLT del flujo de bits codificado y aplicar la modalidad de codificación basada en paleta para decodificar el bloque. En casos en los que hay más de un modo de codificación basado en paleta disponible (por ejemplo, hay más de una técnica basada en paleta disponible para codificación), uno o más elementos

15 sintácticos pueden indicar uno de una pluralidad de modos de paleta diferentes para el bloque.

[0079] En algunos casos, el codificador de vídeo 20 puede codificar un indicador_Modalidad_PLT que es igual a cero para especificar que el bloque actual no está codificado usando una modalidad de paleta. En tales casos, el codificador de vídeo 20 puede codificar el bloque utilizando cualquiera entre varias modalidades de codificación interpredictivas, intrapredictivas y otras. Cuando el indicador_Modalidad_PLT es igual a cero, el codificador de vídeo 20 puede transmitir

25 información adicional (por ejemplo, elementos sintácticos) para indicar la modalidad específica que se utiliza para codificar el bloque respectivo. En algunos ejemplos, como se describe más adelante, la modalidad puede ser una modalidad de codificación HEVC. El uso del indicador_Modalidad_PLT se describe a modo de ejemplo. En otros ejemplos, otros elementos sintácticos, tales como los códigos de múltiples bits, se pueden usar para indicar si la modalidad de codificación

30 basada en paleta se ha de usar para uno o más bloques, o para indicar cuáles, entre una pluralidad de modalidades, se han de usar.

[0080] Cuando se usa una modalidad de codificación basado en paleta, una paleta es transmitida por el codificador de vídeo 20, por ejemplo, utilizando una o más de las técnicas descritas en el presente documento, en el flujo de bits de datos de vídeo codificado, para su uso por el decodificador de vídeo 30. Se puede transmitir una paleta para cada bloque o se puede compartir entre varios bloques. La paleta puede referirse a un número de valores de píxel que son dominantes y/o

35 representativos para el bloque.

[0081] De acuerdo a aspectos de esta divulgación, el tamaño de la paleta, por ejemplo, en términos del número de valores de píxeles que se incluyen en la paleta, puede ser fijo o puede ser señalizado utilizando uno o más elementos sintácticos en un flujo de bits codificado. Como se describe con mayor detalle a continuación, un valor de píxel puede estar compuesto por varias muestras, por ejemplo, según el espacio cromático utilizado para la codificación. Por ejemplo, un valor de píxel puede incluir muestras de luma y crominancia (por ejemplo, luma, crominancia U y crominancia V (YUV) o luma, crominancia Cb y crominancia Cr (YCbCr)). En otro ejemplo, un valor de píxel puede incluir muestras de rojo, verde

40 y azul (RGB). Como se describe en el presente documento, el término valor de píxel puede referirse generalmente a una o más de las muestras que contribuyen a un píxel. Es decir, el término valor de píxel no se refiere necesariamente a todas las muestras que contribuyen a un píxel, y puede usarse para describir un único valor de muestra que contribuye a un píxel.

[0082] De acuerdo a aspectos de esta divulgación, una paleta se puede transmitir por separado para cada componente cromático de un bloque particular. Por ejemplo, en el espacio de color YUV, puede haber una paleta para el componente Y (que representa los valores Y), otra paleta para el componente U (que representa los valores U) y otra paleta más para el componente V (que representa los valores V). En otro ejemplo, una paleta puede incluir todos los componentes de un bloque particular. En este ejemplo, la i-ésima entrada en la paleta puede incluir tres valores (por ejemplo, Yi, Ui, Vi). De

50 acuerdo a los aspectos de esta divulgación, uno o más elementos sintácticos pueden indicar por separado el tamaño de la paleta para cada componente (por ejemplo, Y, U, V o similares). En otros ejemplos, se puede usar un tamaño único

55

para todos los componentes, de manera que uno o más elementos sintácticos indiquen el tamaño de todos los componentes.

[0083] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden realizar la codificación basada en paleta de una manera con pérdidas o sin pérdidas. Es decir, en algunos ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar datos de vídeo sin pérdidas para un bloque usando entradas de paleta que coinciden con los valores de píxel del bloque (o enviando los valores de píxel reales si el valor de píxel no está incluido en la paleta). En otros ejemplos, como se describe con mayor detalle con respecto a la figura 5 a continuación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar datos de vídeo para un bloque utilizando entradas de paleta que no coinciden exactamente con los valores de píxeles del bloque (codificación con pérdidas).

[0084] En algunos ejemplos, las técnicas para la codificación basada en paletas de datos de vídeo pueden usarse con una o más técnicas de codificación diferentes, tales como técnicas para la codificación inter o intrapredictiva. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o decodificador, o un codificador-decodificador combinado (códec), se puede configurar para realizar una codificación interpredictiva e intrapredictiva, así como una codificación basada en paletas.

[0085] La figura 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar las técnicas de esta divulgación. La figura 2 se proporciona con fines de explicación y no debería considerarse limitadora de las técnicas tales como las ampliamente ejemplificadas y descritas en esta divulgación. Con fines de explicación, esta divulgación describe el codificador de vídeo 20 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

[0086] El codificador de vídeo 20 representa un ejemplo de un dispositivo que puede configurarse para realizar técnicas para la codificación de vídeo basada en paletas, de acuerdo a varios ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 20 puede estar configurado para codificar selectivamente varios bloques de datos de vídeo, tales como las CU o PU en la codificación HEVC, usando codificación basada en paletas o codificación no basada en paletas. Las modalidades de codificación no basada en paletas pueden referirse a diversas modalidades de codificación temporal interpredictiva o a modalidades de codificación espacial intrapredictiva, tales como las diversas modalidades de codificación especificadas por el Borrador 10 de la HEVC. El codificador de vídeo 20, en un ejemplo, puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxeles, seleccionar valores de píxeles en una paleta para representar valores de píxeles de al menos algunas posiciones de un bloque de datos de vídeo y señalar información que asocie al menos algunas de las posiciones del bloque de datos de vídeo a entradas en la paleta correspondientes, respectivamente, a los valores de píxeles seleccionados. La información señalizada puede ser utilizada por el decodificador de vídeo 30 para decodificar datos de vídeo.

[0087] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluyen una unidad de procesamiento de predicción 100, una unidad de generación residual 102, una unidad de procesamiento de transformada 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de procesamiento de transformada inversa 110, una unidad de reconstrucción 112, una unidad de filtro 114, una memoria intermedia de imágenes descodificadas 116, y una unidad de codificación por entropía 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de interpredicción 120 y una unidad de procesamiento de intrapredicción 126. La unidad de procesamiento de interpredicción 120 incluye una unidad de estimación de movimiento y una unidad de compensación de movimiento (no mostrada). El codificador de vídeo 20 también incluye una unidad de codificación basada en paletas 122, configurada para realizar diversos aspectos de las técnicas de codificación basada en paletas descritas en esta divulgación. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0088] El codificador de vídeo 20 puede recibir datos de vídeo. El codificador de vídeo 20 puede codificar cada CTU en un fragmento de una imagen de los datos de vídeo. Cada una de las CTU puede asociarse a bloques de árbol de codificación (CTB) de luma de igual tamaño y a los CTB correspondientes de la imagen. Como parte de la codificación de una CTU, la unidad de procesamiento de predicción 100 puede realizar la partición de árbol cuádruple para dividir los CTB de la CTU en bloques progresivamente más pequeños. Los bloques más pequeños pueden ser bloques de codificación de CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un CTB asociado a una CTU en cuatro subbloques de igual tamaño, dividir uno o más de los subbloques en cuatro subsubbloques de igual tamaño, etc.

[0089] El codificador de vídeo 20 puede codificar CU de una CTU para generar representaciones codificadas de las CU (es decir, CU codificadas). Como parte de la codificación de una CU, la unidad de procesamiento de predicción 100 puede dividir los bloques de codificación asociados a la CU entre una o más PU de la CU. Por lo tanto, cada PU puede asociarse a un bloque de predicción de luma y a los correspondientes bloques de predicción de croma. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden dar soporte a las PU de diversos tamaños. Como se ha indicado anteriormente, el tamaño de una CU puede referirse al tamaño del bloque de codificación de luma de la CU, y el tamaño de una PU puede referirse al tamaño de un bloque predictivo de luma de la PU. Suponiendo que el tamaño de una CU particular sea de $2N \times 2N$, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden dar soporte a tamaños de PU de $2N \times 2N$ o $N \times N$ para la intrapredicción, y a tamaños de PU simétricas de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ o similares para la interpredicción. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden dar soporte a la partición asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$, para la predicción inter.

[0090] La unidad de procesamiento de interpredicción 120 puede generar datos predictivos para una PU realizando una interpredicción en cada PU de una CU. Los datos predictivos para la PU pueden incluir bloques predictivos de muestra de la PU e información de movimiento para la PU. La unidad de interpredicción 121 puede realizar diferentes operaciones para una PU de una CU dependiendo de si la PU está en un fragmento I, en un fragmento P o en un fragmento B. En un fragmento I, todas las PU son intrapredichas. Por lo tanto, si la PU está en un fragmento I, la unidad de interpredicción 121 no realiza interpredicción en la PU. Por lo tanto, para bloques codificados en la modalidad I, el bloque predicho se forma usando predicción espacial a partir de bloques contiguos previamente codificados dentro de la misma trama.

[0091] Si una PU está en un fragmento P, la unidad de estimación de movimiento de la unidad de procesamiento de interpredicción 120 puede buscar las imágenes de referencia en una lista de imágenes de referencia (por ejemplo, "ListalmagenRef0") para una región de referencia para la PU. La región de referencia para la PU puede ser una región, dentro de una imagen de referencia, que contiene bloques de muestra que se corresponden más estrechamente con los bloques de muestra de la PU. La unidad de estimación de movimiento 122 puede generar un índice de referencia que indique una posición en la ListalmagenRef0 de la imagen de referencia que contenga la región de referencia para la PU. Además, la unidad de estimación de movimiento puede generar un vector de movimiento que indique un desplazamiento espacial entre un bloque de codificación de la PU y una ubicación de referencia asociada a la región de referencia. Por ejemplo, el vector de movimiento puede ser un vector bidimensional que proporcione un desplazamiento entre las coordenadas en la imagen decodificada actual y las coordenadas en una imagen de referencia. La unidad de estimación de movimiento puede emitir el índice de referencia y el vector de movimiento como la información de movimiento de la PU. La unidad de compensación de movimiento de la unidad de procesamiento de interpredicción 120 puede generar los bloques predictivos de la PU basándose en muestras reales o interpoladas en la ubicación de referencia, indicada por el vector de movimiento de la PU.

[0092] Si una PU está en un fragmento B, la unidad de estimación de movimiento puede realizar unipredicción o bipredicción para la PU. Para realizar la unipredicción para la PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia de la ListalmagenRef0 o de una segunda lista de imágenes de referencia ("ListalmagenRef1") para una región de referencia para la PU. La unidad de estimación de movimiento puede emitir, como la información de movimiento de la PU, un índice de referencia que indique una posición en la ListalmagenRef0 o en la ListalmagenRef1 de la imagen de referencia que contiene la región de referencia, un vector de movimiento que indique un desplazamiento espacial entre un bloque predictivo de la PU y una ubicación de referencia asociada a la región de referencia, y uno o más indicadores de dirección de predicción que indiquen si la imagen de referencia está en la ListalmagenRef0 o en la ListalmagenRef1. La unidad de compensación de movimiento de la unidad de procesamiento de interpredicción 120 puede generar los bloques predictivos de la PU basándose, al menos en parte, en muestras reales o interpoladas en la región de referencia indicada por el vector de movimiento de la PU.

[0093] Para realizar la interpredicción bidireccional para una PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia en la ListalmagenRef0 para una región de referencia para la PU y también puede buscar las imágenes de referencia en la ListalmagenRef1 para otra región de referencia para la PU. La unidad de estimación de movimiento puede generar índices de imágenes de referencia que indiquen posiciones en la ListalmagenRef0 y en la ListalmagenRef1 de las imágenes de referencia que contienen las regiones de referencia. Además, la unidad de estimación de movimiento puede generar vectores de movimiento que indiquen desplazamientos espaciales entre la ubicación de referencia asociada a las regiones de referencia y un bloque de muestra de la PU. La información de movimiento de la PU puede incluir los índices de referencia y los MV de la PU. La unidad de compensación de movimiento puede generar los bloques predictivos de muestra de la PU basándose, al menos en parte, en muestras reales o interpoladas en las regiones de referencia indicadas por los vectores de movimiento de la PU.

[0094] De acuerdo con diversos ejemplos de esta divulgación, el codificador de vídeo 20 puede configurarse para realizar una codificación basada en paletas. Con respecto al marco de la HEVC, como ejemplo, las técnicas de codificación basada en paletas se pueden configurar para usarse como un modo de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basada en paletas pueden configurarse para usarse como una modalidad de PU en el entorno de la HEVC. En consecuencia, todos los procesos divulgados, descritos en el presente documento (a lo largo de esta divulgación) en el contexto de una modalidad de CU, pueden, adicionalmente o alternativamente, aplicarse a las PU. Sin embargo, estos ejemplos basados en la HEVC no deberían considerarse como una restricción o limitación de las técnicas de codificación basada en paletas, descritas en el presente documento, ya que dichas técnicas se pueden aplicar para que funcionen independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación por paleta pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[0095] La unidad de codificación basada en paletas 122, por ejemplo, puede realizar una decodificación basada en paletas cuando se seleccione una modalidad de codificación basada en paletas, por ejemplo, para una CU o PU. Por ejemplo, la unidad de codificación basada en paletas 122 puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxel, seleccionar valores de píxel en una paleta para representar valores de píxeles de al menos algunas posiciones de un bloque de datos de vídeo y señalar información que asocie al menos algunas de las posiciones del bloque de datos de vídeo a entradas en la paleta correspondientes, respectivamente, a los valores de píxel seleccionados. Aunque se describen diversas funciones como realizadas por la unidad de codificación basada en paletas 122, algunas de, o todas, estas funciones pueden realizarse por otras unidades de procesamiento, o una combinación de diferentes unidades de procesamiento.

[0096] La unidad de codificación basada en paletas 122 puede estar configurada para generar cualquiera de los diversos elementos sintácticos descritos en el presente documento. Por consiguiente, el codificador de vídeo 20 puede configurarse para codificar bloques de datos de vídeo usando modos de código basados en paletas como se describe en esta divulgación. El codificador de vídeo 20 puede codificar selectivamente un bloque de datos de vídeo usando un modo de codificación por paleta, o codificar un bloque de datos de vídeo usando un modo diferente, por ejemplo, un modo de codificación HEVC interpredictiva o intrapredictiva. El bloque de datos de vídeo puede ser, por ejemplo, una CU o PU generada de acuerdo con un proceso de codificación de HEVC. Un codificador de vídeo 20 puede codificar algunos bloques con modos de predicción temporal interpredictiva o codificación espacial intrapredictiva y decodificar otros bloques con el modo de codificación basado en paleta.

[0097] La unidad de procesamiento de intrapredicción 126 puede generar datos predictivos para una PU realizando una intrapredicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos para la PU y diversos elementos sintácticos. La unidad de procesamiento de intrapredicción 126 puede realizar la intrapredicción en las PU en fragmentos I, fragmentos P y fragmentos B.

[0098] Para realizar la intrapredicción en una PU, la unidad de procesamiento de intrapredicción 126 puede usar múltiples modalidades de intrapredicción para generar múltiples conjuntos de datos predictivos para la PU. Para usar una modalidad de intrapredicción para generar un conjunto de datos predictivos para la PU, la unidad de procesamiento de intrapredicción 126 puede extender muestras a partir de bloques de muestra de PU contiguas, entre los bloques de muestras de la PU en una dirección asociada a la modalidad de intrapredicción. Las PU adyacentes pueden estar arriba, arriba y a la derecha, arriba y a la izquierda, o a la izquierda de la PU, suponiendo un orden de codificación de izquierda a derecha, de arriba a abajo, para las PU, CU y CTU. La unidad de procesamiento de predicción intra 126 puede usar diversos números de modos de predicción intra, por ejemplo, 33 modos de predicción intra direccional. En algunos ejemplos, el número de modalidades de intrapredicción puede depender del tamaño de la región asociada a la PU.

[0099] La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para las PU de una CU a partir de los datos predictivos generados por la unidad de procesamiento de interpredicción 120 para las PU o los datos predictivos generados por la unidad de procesamiento de intrapredicción 126 para las PU. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos predictivos para las PU de la CU basándose en las mediciones de velocidad/distorsión de los conjuntos de datos predictivos. Los bloques de muestras predictivas de los datos predictivos seleccionados pueden denominarse en el presente documento como los bloques predictivos de muestra seleccionados.

[0100] La unidad de generación residual 102 puede generar, basándose en los bloques de codificación de luma, Cb y Cr de una CU y los bloques predictivos de luma, Cb y Cr seleccionados de las PU de la CU, y bloques residuales de luma,

Cb y Cr de la CU. Por ejemplo, la unidad de generación residual 102 puede generar los bloques residuales de la CU de modo que cada muestra en los bloques residuales tenga un valor igual a una diferencia entre una muestra en un bloque de codificación de la CU y una muestra correspondiente en un correspondiente bloque predictivo seleccionado de una PU de la CU.

[0101] La unidad de procesamiento de transformada 104 puede realizar la partición de árbol cuádruple para particionar los bloques residuales de una CU en bloques de transformada asociados a las TU de la CU. De este modo, una TU puede estar asociada a un bloque de transformación de luma y a dos bloques de transformación de croma. Los tamaños y las posiciones de los bloques de transformación de luma y croma de las TU de una CU pueden o no basarse en los tamaños y posiciones de los bloques de predicción de las PU de la CU. Una estructura de árbol cuádruple conocida como "árbol cuádruple residual" (RQT) puede incluir nodos asociados con cada una de las regiones. Las TU de una CU pueden corresponder a nodos de hoja del RQT.

[0102] La unidad de procesamiento de transformada 104 puede generar bloques de coeficientes de transformada para cada TU de una CU aplicando una o más transformadas a los bloques de transformada de la TU. La unidad de procesamiento de transformada 104 puede aplicar diversas transformadas a un bloque de transformada asociado a una TU. Por ejemplo, la unidad de procesamiento de transformada 104 puede aplicar una transformada de coseno discreta (DCT), una transformada direccional o una transformada conceptualmente similar al bloque de transformada. En algunos ejemplos, la unidad de procesamiento de transformada 104 no aplica transformadas a un bloque de transformada. En dichos ejemplos, el bloque de transformada puede tratarse como un bloque de coeficientes de transformada.

[0103] La unidad de cuantización 106 puede cuantizar los coeficientes de transformada en un bloque de coeficientes. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes de transformación. Por ejemplo, un coeficiente de transformada de n bits puede redondearse hacia abajo hasta un coeficiente de transformada de m bits durante la cuantificación, donde n es mayor que m . La unidad de cuantificación 106 puede cuantificar un bloque de coeficientes asociado a una TU de una CU basándose en un valor de parámetro de cuantificación (QP) asociado a la CU. El codificador de vídeo 20 puede ajustar el grado de cuantificación aplicado a los bloques de coeficientes asociados a una CU, ajustando el valor del QP asociado a la CU. La cuantificación puede introducir pérdida de información, por lo que los coeficientes de transformada cuantificados pueden tener una precisión inferior a los originales.

[0104] La unidad de cuantización inversa 108 y la unidad de procesamiento de transformada inversa 110 pueden aplicar la cuantización inversa y transformadas inversas al bloque de coeficientes, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficientes. La unidad de reconstrucción 112 puede añadir el bloque residual reconstruido a las muestras correspondientes de uno o más bloques predictivos generados por la unidad de procesamiento de predicción 100 para producir un bloque de transformada reconstruido asociado con una TU. Mediante la reconstrucción de bloques de transformada para cada TU de una CU de esta manera, el codificador de vídeo 20 puede reconstruir los bloques de codificación de la CU.

[0105] La unidad de filtro 114 puede realizar una o más operaciones de desbloqueo para reducir las distorsiones de bloqueo en los bloques de codificación asociados con una CU. La memoria intermedia de imágenes decodificadas 116 puede almacenar los bloques de codificación reconstruidos después de que la unidad de filtro 114 realice las una o más operaciones de desbloqueo en los bloques de codificación reconstruidos. La unidad de procesamiento de predicción inter 120 puede usar una imagen de referencia que contenga los bloques de codificación reconstruidos para realizar la predicción inter en las PU de otras imágenes. Además, la unidad de procesamiento de predicción intra 126 puede usar bloques de codificación reconstruidos en la memoria intermedia de imágenes descodificadas 116 para realizar predicción intra en otras PU en la misma imagen que la CU.

[0106] La unidad de codificación por entropía 118 puede recibir datos desde otros componentes funcionales del codificador de vídeo 20. Por ejemplo, la unidad de codificación por entropía 118 puede recibir bloques de coeficientes desde la unidad de cuantización 106 y puede recibir elementos sintácticos desde la unidad de procesamiento de predicción 100. La unidad de codificación por entropía 118 puede realizar una o más operaciones de codificación por entropía en los datos para generar datos codificados por entropía. Por ejemplo, la unidad de codificación por entropía 118 puede realizar una operación de codificación de longitud variable adaptable al contexto (CAVLC), una operación de CABAC, una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética binaria adaptable al contexto basada en sintaxis (SBAC), una operación de codificación por entropía de partición de intervalos de probabilidad (PIPE), una operación de codificación Exponencial-Golomb u otro tipo de operación de codificación por

entropía en los datos. El codificador de vídeo 20 puede emitir un flujo de bits que incluya datos codificados por entropía, generados por la unidad de codificación por entropía 118. Por ejemplo, el flujo de bits puede incluir datos que representen un RQT para una CU.

5 **[0107]** En algunos ejemplos, la codificación residual no se realiza con la codificación de paleta. Por consiguiente, el codificador de vídeo 20 no puede realizar la transformada o la cuantización cuando se codifique usando un modo de codificación de paleta. Además, el codificador de vídeo 20 puede codificar por entropía los datos generados usando un modo de codificación de paleta independientemente de los datos residuales.

10 **[0108]** La FIG. 3 es un diagrama de bloques que ilustra un decodificador de vídeo 30 ejemplar que está configurado para implementar las técnicas de esta divulgación. La FIG. 3 se proporciona con fines de explicación y no se limita a las técnicas como las ampliamente ejemplificadas y descritas en esta divulgación. Con fines de explicación, esta divulgación describe el decodificador de vídeo 30 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

15 **[0109]** El codificador de vídeo 30 representa un ejemplo de un dispositivo que puede configurarse para realizar técnicas para la codificación de vídeo basada en paletas, de acuerdo a varios ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 30 puede estar configurado para decodificar selectivamente varios bloques de datos de vídeo, tales como las CU o las PU en la codificación HEVC, usando codificación basada en paletas o bien codificación no basada en paletas. Las modalidades de codificación no basada en paletas pueden referirse a diversas modalidades de codificación temporal interpredictiva o a modalidades de codificación espacial intrapredictiva, tales como las diversas modalidades de codificación especificadas por el Borrador 10 de la HEVC. El decodificador de vídeo 30, en un ejemplo, puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxel, recibir información de recepción que asocie al menos algunas ubicaciones de un bloque de datos de vídeo con entradas en la paleta, seleccionar valores de píxeles en la paleta en función de la información y reconstruir los valores de píxel del bloque en función de los valores de píxeles seleccionados.

20 **[0110]** En el ejemplo de la FIG. 3, el decodificador de vídeo 30 incluye una unidad de descodificación por entropía 150, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de procesamiento de transformada inversa 156, una unidad de reconstrucción 158, una unidad de filtro 160 y una memoria intermedia de imágenes descodificadas 162. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 164 y una unidad de procesamiento de intrapredicción 166. El decodificador de vídeo 30 también incluye una unidad de decodificación basada en paletas 165 configurada para realizar diversos aspectos de las técnicas de codificación basada en paletas descritas en esta divulgación. En otros ejemplos, el decodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

30 **[0111]** Una memoria intermedia de imágenes codificadas (CPB) puede recibir y almacenar datos de vídeo codificados (por ejemplo, unidades de NAL) de un flujo de bits. La unidad de decodificación por entropía 150 puede recibir datos de vídeo codificado (por ejemplo, unidades de NAL) desde el CPB 151 y analizar sintácticamente las unidades de NAL para decodificar elementos sintácticos. La unidad de decodificación por entropía 150 puede decodificar por entropía elementos sintácticos codificados por entropía en las unidades de NAL. La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de procesamiento de transformada inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 160 pueden generar datos de vídeo descodificados basándose en los elementos sintácticos extraídos del flujo de bits.

45 **[0112]** Las unidades NAL del flujo de bits pueden incluir unidades NAL de fragmentos codificados. Como parte de la decodificación del flujo de bits, la unidad de decodificación por entropía 150 puede extraer y decodificar por entropía elementos sintácticos de las unidades de NAL de fragmentos codificados. Cada uno de los fragmentos codificados puede incluir una cabecera de fragmento y datos de fragmento. La cabecera de fragmento puede contener elementos sintácticos pertenecientes a un fragmento. Los elementos sintácticos en la cabecera de fragmento pueden incluir un elemento sintáctico que identifique un PPS asociado a una imagen que contenga el fragmento.

50 **[0113]** Además de decodificar los elementos sintácticos del flujo de bits, el decodificador de vídeo 30 puede realizar una operación de reconstrucción en una CU no dividida. Para realizar la operación de reconstrucción en una CU no dividida, el decodificador de vídeo 30 puede realizar una operación de reconstrucción en cada TU de la CU. Realizando la operación de reconstrucción para cada TU de la CU, el decodificador de vídeo 30 puede reconstruir bloques residuales de la CU.

[0114] Como parte de la realización de una operación de reconstrucción en una TU de una CU, la unidad de cuantización inversa 154 puede cuantizar de forma inversa, es decir, descuantizar, los bloques de coeficientes asociados a la TU. La unidad de cuantización inversa 154 puede utilizar un valor de QP asociado a la CU de la TU para determinar un grado de cuantización y, asimismo, un grado de cuantización inversa, para aplicar por la unidad de cuantización inversa 154. Es decir, la relación de compresión, es decir, la relación entre el número de bits usados para representar la secuencia original y la comprimida, puede controlarse ajustando el valor del QP usado al cuantificar los coeficientes de transformada. La relación de compresión también puede depender del procedimiento de codificación por entropía empleado.

[0115] Después de que la unidad de cuantización inversa 154 cuantiza inversamente un bloque de coeficientes, la unidad de procesamiento de transformada inversa 156 puede aplicar una o más transformadas inversas al bloque de coeficientes para generar un bloque de muestras residual asociado a la TU. Por ejemplo, la unidad de procesamiento de transformada inversa 156 puede aplicar una DCT inversa, una transformada entera inversa, una transformada de Karhunen-Loeve (KLT) inversa, una transformada de rotación inversa, una transformada direccional inversa u otra transformada inversa al bloque de coeficientes.

[0116] Si se codifica una PU usando la intrapredicción, la unidad de procesamiento de intrapredicción 166 puede realizar la intrapredicción para generar bloques predictivos para la PU. La unidad de procesamiento de intrapredicción 166 puede usar una modalidad de intrapredicción para generar los bloques predictivos de luma, Cb y Cr para la PU basándose en los bloques de predicción de las PU espacialmente contiguas. La unidad de procesamiento de predicción intra 166 puede determinar el modo de predicción intra para la PU basándose en uno o más elementos sintácticos descodificados a partir del flujo de bits.

[0117] La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (ListImagenRef0) y una segunda lista de imágenes de referencia (ListImagenRef1) basándose en los elementos sintácticos extraídos del flujo de bits. Además, si se codifica una PU usando la interpredicción, la unidad de decodificación por entropía 150 puede extraer información de movimiento para la PU. La unidad de compensación de movimiento 164 puede determinar, basándose en la información de movimiento de la PU, una o más regiones de referencia para la PU. La unidad de compensación de movimiento 164 puede generar, basándose en bloques de muestras en los uno o más bloques de referencia para la PU, bloques predictivos de luma, Cb y Cr para la PU.

[0118] La unidad de reconstrucción 158 puede utilizar los bloques de transformación de luma, Cb y Cr, asociados a las TU de una CU y los bloques predictivos de luma, Cb y Cr de las PU de la CU, es decir, datos de intrapredicción o bien datos de inter-predicción, según corresponda, para reconstruir los bloques de codificación de luma, Cb y Cr de la CU. Por ejemplo, la unidad de reconstrucción 158 puede agregar muestras de los bloques de transformación de luma, Cb y Cr a las muestras correspondientes de los bloques predictivos de luma, Cb y Cr para reconstruir los bloques de codificación de luma, Cb y Cr de la CU.

[0119] La unidad de filtro 160 puede realizar una operación de desbloqueo para reducir las distorsiones de bloqueo asociadas a los bloques de codificación de luma, Cb y Cr de la CU. El decodificador de vídeo 30 puede almacenar los bloques de codificación de luma, Cb y Cr de la CU en la memoria intermedia de imágenes decodificadas 162. La memoria intermedia de imágenes decodificadas 162 puede proporcionar imágenes de referencia para la posterior compensación de movimiento, la intrapredicción y la presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Por ejemplo, el decodificador de vídeo 30 puede realizar, basándose en los bloques de luma, Cb y Cr en la memoria intermedia de imágenes decodificadas 162, operaciones de intrapredicción o de interpredicción en PU de otras CU. De esta forma, el decodificador de vídeo 30 puede extraer, a partir del flujo de bits, los niveles de coeficientes de transformada del bloque de coeficientes luma significativos, cuantificar inversamente los niveles de coeficientes de transformada, aplicar una transformada a los niveles de coeficientes de transformada para generar un bloque de transformada, generar, basándose, al menos en parte, en el bloque de transformada, un bloque de codificación, y emitir el bloque de codificación para su visualización.

[0120] De acuerdo con diversos ejemplos de esta divulgación, el codificador de vídeo 30 puede configurarse para realizar una codificación basada en paletas. La unidad de codificación basada en paletas 165, por ejemplo, puede realizar una decodificación basada en paletas cuando se seleccione una modalidad de codificación basada en paletas, por ejemplo, para una CU o PU. Por ejemplo, la unidad de decodificación basada en paletas 165 puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxeles, recibir información que asocie al menos algunas ubicaciones de un bloque de datos de vídeo a entradas en la paleta, seleccionar valores de píxeles en la paleta en función de la información y reconstruir valores de píxel del bloque en función de los valores de píxel seleccionados. Aunque se describen

diversas funciones como realizadas por la unidad de decodificación basada en paletas 165, algunas de, o todas, estas funciones pueden realizarse por otras unidades de procesamiento, o por una combinación de diferentes unidades de procesamiento.

5 **[0121]** La unidad de decodificación basada en paletas 165 puede recibir información de modalidad de codificación por paleta, y realizar las operaciones anteriores cuando la información de modalidad de codificación de paleta indique que la modalidad de codificación por paleta se aplica al bloque. Cuando la información del modo de codificación por paleta indica que el modo de codificación por paleta no se aplica al bloque, o cuando la información de otro modo indica el uso de un modo diferente, la unidad de decodificación basada en paletas 165 decodifica el bloque de datos de vídeo usando un modo de codificación no basada en paletas, por ejemplo, dicho modo de codificación HEVC interpredictivo o intrapredictivo, cuando la información del modo de codificación por paleta indica que el modo de codificación de la paleta no se aplica al bloque. El bloque de datos de vídeo puede ser, por ejemplo, una CU o PU generada de acuerdo con un proceso de codificación de HEVC. Un codificador de vídeo 30 puede codificar algunos bloques con modos de predicción temporal interpredictiva o codificación espacial intrapredictiva y decodificar otros bloques con el modo de codificación basada en paletas. El modo de codificación basada en paletas puede comprender uno de una pluralidad de diferentes modos de codificación basada en paletas, o puede haber un único modo de codificación basada en paletas.

20 **[0122]** La información de modalidad de codificación por paleta, recibida por la unidad de decodificación basada en paletas 165, puede comprender un elemento sintáctico de modalidad de paleta, tal como un indicador. Un primer valor del elemento sintáctico de modalidad de paleta indica que la modalidad de codificación por paleta se aplica al bloque y un segundo valor del elemento sintáctico de modalidad de paleta indica que la modalidad de codificación por paleta no se aplica al bloque de datos de vídeo. La unidad de decodificación basada en paletas 165 puede recibir (por ejemplo, desde el codificador de vídeo 20) la información de modalidad de codificación de paleta en uno o más entre un nivel de unidad predictiva, un nivel de unidad de codificación, un nivel de fragmento o un nivel de imagen, o puede recibir una indicación de si la modalidad de codificación de paleta está habilitada en al menos uno entre el conjunto de parámetros de imagen (PPS), el conjunto de parámetros de secuencia (SPS) o el conjunto de parámetros de vídeo (VPS).

30 **[0123]** En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede deducir la información de modalidad de codificación de paleta en función de uno o más entre un tamaño del bloque de codificación, un tipo de trama, un espacio cromático, un componente cromático, un tamaño de trama, una velocidad de trama, un identificador de capa en la codificación de vídeo ajustable a escala o un identificador de vista en la codificación de múltiples vistas asociada al bloque de datos de vídeo.

35 **[0124]** La unidad de decodificación basada en paletas 165 también puede configurarse para recibir información que define al menos algunas de las entradas en la paleta con datos de vídeo, y generar la paleta basándose, al menos en parte, en la información recibida. El tamaño de la paleta puede ser fijo o variable. En algunos casos, el tamaño de la paleta es variable y se puede ajustar en función de la información señalizada con los datos de vídeo. La información señalizada puede especificar si una entrada en la paleta es una última entrada en la paleta. Además, en algunos casos, la paleta puede tener un tamaño máximo. El tamaño de la paleta también se puede transmitir o deducir condicionalmente. Las condiciones pueden ser el tamaño de la CU, el tipo de trama, el espacio de color, el componente de color, el tamaño de la trama, la velocidad de trama, el identificador de capa en la codificación de vídeo ajustable a escala o el identificador de vista en la codificación de múltiples vistas.

45 **[0125]** La paleta puede ser una paleta única que incluya entradas que indiquen valores de píxel para un componente de luma y componentes de croma del bloque. En este caso, cada entrada en la paleta es una entrada triple que indica valores de píxel para el componente luma y dos componentes de croma. Alternativamente, la paleta incluye una paleta de luma que incluye entradas que indican valores de píxel de un componente de luma del bloque, y paletas de croma que incluyen entradas que indican valores de píxel para los respectivos componentes de croma del bloque.

50 **[0126]** En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede generar la paleta al predecir las entradas en la paleta en función de los datos procesados previamente. Los datos procesados previamente pueden incluir paletas, o información de paletas, para bloques vecinos previamente decodificados. La unidad de decodificación basada en paletas 165 puede recibir un elemento sintáctico de predicción que indica si las entradas en la paleta han de predecirse. El elemento sintáctico de predicción puede incluir una pluralidad de elementos sintácticos de predicción que indican, respectivamente, si se han de predecir las entradas en las paletas para componentes de luma y croma.

[0127] Con respecto a una paleta predictiva, por ejemplo, una paleta predictiva puede contener entradas de paleta de uno o más bloques vecinos, incluidos los bloques espacialmente vecinos y/o los bloques vecinos en un orden de exploración particular de los bloques. En un ejemplo, los bloques vecinos se pueden ubicar espacialmente a la izquierda (bloque vecino izquierdo) o arriba (bloque vecino superior) del bloque que se está codificando actualmente. En otro ejemplo, la unidad de decodificación basada en paletas 165 puede determinar las entradas de paleta predictivas utilizando los valores de muestra más frecuentes en un vecino casual del bloque actual. En otro ejemplo, los bloques vecinos pueden colindar con el bloque que se está codificando actualmente, de acuerdo a un orden de exploración particular utilizado para codificar los bloques. Es decir, los bloques vecinos pueden ser uno o más bloques codificados antes del bloque actual en el orden de exploración. La unidad de decodificación basada en paletas 165 puede decodificar uno o más elementos sintácticos para indicar la ubicación de los bloques vecinos desde los que se copian las una o más paletas.

[0128] Así, en un ejemplo, la unidad de decodificación basada en paletas 165 puede, en algunos ejemplos, predecir al menos algunas de las entradas en la paleta basándose en las entradas en una paleta para un bloque vecino izquierdo o un bloque vecino superior en un fragmento o imagen. En este caso, las entradas en la paleta que se predicen en función de las entradas en una paleta para el bloque vecino izquierdo o bien el bloque vecino superior pueden predecirse mediante la unidad de decodificación basada en paletas 165, en función de un elemento sintáctico que indica la selección del bloque izquierdo vecino o el bloque vecino superior para su predicción. El elemento sintáctico puede ser un indicador que tiene un valor que indica la selección del bloque vecino izquierdo o el bloque vecino superior para su predicción.

[0129] En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede recibir uno o más elementos sintácticos de predicción que indican si al menos algunas entradas seleccionadas en la paleta, han de predecirse, entrada por entrada, y generar las entradas en consecuencia. Por ejemplo, la unidad de decodificación basada en paletas 165 puede decodificar uno o más elementos sintácticos para indicar, para cada entrada de una paleta predictiva, si la entrada de la paleta está incluida en la paleta para el bloque actual. Si no se predice una entrada, la unidad de decodificación basada en paletas 165 puede decodificar uno o más elementos sintácticos adicionales para especificar las entradas no predichas, así como el número de dichas entradas. Por lo tanto, la unidad de decodificación basada en paletas 165 puede predecir algunas de las entradas y recibir información que especifique directamente otras entradas en la paleta, incluido el número de entradas adicionales.

[0130] En algunos ejemplos, las técnicas para predecir una paleta completa se pueden combinar con técnicas para predecir una o más entradas de una paleta. Por ejemplo, la unidad de decodificación basada en paletas 165 puede decodificar uno o más elementos sintácticos en un flujo de bits para indicar si la paleta actual está totalmente copiada a partir de la paleta predictiva. Si este no es el caso, la unidad de decodificación basada en paletas 165 puede decodificar uno o más elementos sintácticos en un flujo de bits para indicar si se ha copiado cada entrada en la paleta predictiva.

[0131] En otro ejemplo, en lugar de recibir el número de entradas y el valor de paleta, la unidad de decodificación basada en paletas 165 puede recibir, después de cada valor de paleta, un indicador para indicar si el valor de la paleta señalizada es la entrada final de paleta para la paleta. La unidad de decodificación basada en paletas 165 puede no recibir tal indicador de "fin de paleta" si la paleta ya ha alcanzado un cierto tamaño máximo.

[0132] La información, recibida por la unidad de decodificación basada en paletas 165, que asocia al menos algunas posiciones de un bloque de datos de vídeo a entradas en la paleta, puede incluir información de correlación que incluye valores de índice de paleta para al menos algunas de las posiciones en el bloque, en donde cada uno de los valores de índices de paleta corresponde a una de las entradas en la paleta. La información de correlación puede incluir uno o más elementos sintácticos de racha, cada uno de los cuales indica un número de posiciones consecutivas en el bloque que tienen el mismo valor de índice de paleta.

[0133] En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede recibir información que indica la copia de líneas, según la cual los valores de píxel o índice para una línea de posiciones en el bloque se copian de los valores de píxel o índice para otra línea de posiciones en el bloque. La unidad de decodificación basada en paletas 165 puede usar esta información para realizar copias de líneas para determinar los valores de píxeles o las entradas en la paleta para varias posiciones de un bloque. La línea de posiciones puede comprender una fila, una parte de una fila, una columna o una parte de una columna de posiciones del bloque.

[0134] La unidad de decodificación basada en paletas 165 puede generar la paleta en parte al recibir valores de píxeles para una o más posiciones del bloque, y agregar los valores de píxeles a las entradas en la paleta para generar dinámicamente al menos una parte de la paleta sobre la marcha. Agregar los valores de píxel puede comprender agregar

los valores de píxel a una paleta inicial que comprende un conjunto inicial de entradas, o a una paleta vacía que no incluye un conjunto inicial de entradas. En algunos ejemplos, agregar comprende agregar los valores de píxeles para agregar nuevas entradas a una paleta inicial que comprende un conjunto inicial de entradas o completar entradas existentes en la paleta inicial, o reemplazar o cambiar los valores de píxeles de las entradas en la paleta inicial.

[0135] En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede determinar un tamaño máximo fijo para una paleta. Al alcanzar el tamaño máximo, la unidad de decodificación basada en paletas 165 puede eliminar una o más entradas de la paleta. En un ejemplo, la unidad de decodificación basada en paletas 165 puede eliminar la entrada más antigua de la paleta, por ejemplo, utilizando una cola FIFO. En otro ejemplo, la unidad de decodificación basada en paletas 165 puede eliminar la entrada menos utilizada. En otro ejemplo más, la unidad de decodificación basada en paletas 165 puede tomar una determinación ponderada con respecto a qué entrada eliminar, en función de cuándo se agregó a la paleta una entrada candidata a ser eliminada y del uso relativo de esa entrada.

[0136] En algunos ejemplos, la paleta puede ser una paleta cuantizada en la que un valor de píxel seleccionado de la paleta para una de las posiciones en el bloque es diferente a un valor de píxel real de la posición en el bloque, de tal manera que el proceso de decodificación tenga pérdidas. Por ejemplo, el mismo valor de píxel puede ser seleccionado desde la paleta para dos posiciones diferentes que tienen diferentes valores de píxel reales.

[0137] La FIG. 4 es un diagrama conceptual que ilustra un ejemplo de determinación de una paleta para codificar datos de vídeo, congruente con las técnicas de esta divulgación. El ejemplo de la FIG. 4 incluye una imagen 178 que tiene una primera unidad de codificación (CU) 180 que está asociada a las primeras paletas 184 y una segunda CU 188 que está asociada a las segundas paletas 192. Como se describe en mayor detalle a continuación y de acuerdo a las técnicas de esta divulgación, las segundas paletas 192 se basan en las primeras paletas 184. La imagen 178 también incluye el bloque 196 codificado con una modalidad de codificación de intrapredicción y el bloque 200 que está codificado con una modalidad de codificación de inter-predicción.

[0138] Las técnicas de la FIG. 4 se describen en el contexto del codificador de vídeo 20 (FIG. 1 y en la FIG. 2) y el decodificador de vídeo 30 (FIG. 1 y en la FIG. 3) y con respecto a la norma de codificación de vídeo HEVC, con fines de explicación. Sin embargo, debería entenderse que las técnicas de esta divulgación no están limitadas de esta manera, y que pueden ser aplicadas por otros procesadores y/o dispositivos de codificación de vídeo en otros procesos y/o normas de codificación de vídeo.

[0139] En general, una paleta se refiere a un número de valores de píxel que son dominantes y / o representativos para una CU actualmente codificada, la CU 188 en el ejemplo de la FIG. 4. Las primeras paletas 184 y las segundas paletas 192 se muestran incluyendo múltiples paletas. En algunos ejemplos, de acuerdo a aspectos de esta divulgación, un codificador de vídeo (tal como el codificador de vídeo 20 o el decodificador de vídeo 30) puede codificar paletas por separado para cada componente cromático de una CU. Por ejemplo, el codificador de vídeo 20 puede codificar una paleta para un componente de luma (Y) de una CU, otra paleta para un componente de croma (U) de la CU y otra paleta más para el componente de croma (V) de la CU. En este ejemplo, las entradas de la paleta Y pueden representar valores Y de píxeles de la CU, las entradas de la paleta U pueden representar valores U de píxeles de la CU y las entradas de la paleta V pueden representar valores V de píxeles de la CU.

[0140] En otros ejemplos, el codificador de vídeo 20 puede codificar una única paleta para todos los componentes de color de una CU. En este ejemplo, el codificador de vídeo 20 puede codificar una paleta que tiene una i-ésima entrada que es un valor triple, incluyendo Yi, Ui y Vi. En este caso, la paleta incluye valores para cada uno de los componentes de los píxeles. En consecuencia, la representación de las paletas 184 y 192 como un conjunto de paletas que tienen múltiples paletas individuales es simplemente un ejemplo y no pretende ser limitador.

[0141] En el ejemplo de la FIG. 4, cada una de las primeras paletas 184 incluye tres entradas 202 a 206 que tienen un valor de índice de entrada 1, un valor de índice de entrada 2 y un valor de índice de entrada 3, respectivamente. Las entradas 202 a 206 relacionan los valores de índice con los valores de píxel, incluyendo el valor de píxel A, el valor de píxel B y el valor de píxel C, respectivamente. Como se describe en el presente documento, en vez de codificar los valores de píxel reales de la primera CU 180, un codificador de vídeo (tal como un codificador de vídeo 20 o un decodificador de vídeo 30) puede usar la codificación basada en paletas para codificar los píxeles del bloque usando los índices 1 a 3. Es decir, para cada posición de píxel de la primera CU 180, el codificador de vídeo 20 puede codificar un valor de índice para el píxel, donde el valor de índice está asociado a un valor de píxel en una o más de las primeras paletas 184. El decodificador de vídeo 30 puede obtener los valores de índice desde un flujo de bits y reconstruir los valores de píxel

usando los valores de índice y una o más de las primeras paletas 184. De este modo, las primeras paletas 184 son transmitidas por el codificador de vídeo 20 en un flujo de bits de datos de vídeo codificado para su uso por el decodificador de vídeo 30 en la decodificación basada en paletas. En general, se pueden transmitir una o más paletas para cada CU o se pueden compartir entre diferentes CU.

[0142] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar segundas paletas 192 basándose en las primeras paletas 184. Por ejemplo, el codificador de vídeo 20 puede codificar un indicador_paleta_pred para cada CU (incluyendo, como ejemplo, la segunda CU 188) para indicar si la paleta para la CU se predice a partir de una o más paletas asociadas a otras una o más CU, tales como las CU vecinas (espacialmente o en función del orden de exploración) o las muestras más frecuentes de un vecino casual. Por ejemplo, cuando el valor de tal indicador es igual a uno, el decodificador de vídeo 30 puede determinar que las segundas paletas 192 para la segunda CU 188 se pronostican a partir de una o más paletas ya decodificadas y, por lo tanto, no se incluyen nuevas paletas para la segunda CU 188 en un flujo de bits que contiene el indicador_paleta_pred. Cuando dicho indicador es igual a cero, el decodificador de vídeo 30 puede determinar que las paletas 192 para la segunda CU 188 están incluidas en el flujo de bits como una nueva paleta. En algunos ejemplos, el indicador_paleta_pred puede codificarse por separado para cada componente cromático diferente de una CU (por ejemplo, tres indicadores, uno para Y, uno para U y uno para V, para una CU en vídeo YUV). En otros ejemplos, un solo indicador_paleta_pred puede codificarse para todos los componentes cromáticos de una CU.

[0143] En el ejemplo anterior, el indicador_paleta_pred se señala por cada CU para indicar si se predice alguna de las entradas de la paleta para el bloque actual. En algunos ejemplos, uno o más elementos sintácticos pueden ser señalizados por cada entrada. Es decir, se puede señalar un indicador para cada entrada de un predictor de paleta para indicar si esa entrada está presente en la paleta actual. Como se ha indicado anteriormente, si una entrada de la paleta no se predice, la entrada de la paleta se puede señalar explícitamente.

[0144] Cuando se determinan las segundas paletas 192 en relación con las primeras paletas 184 (por ejemplo, el indicador_paleta_pred es igual a uno), el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden ubicar uno o más bloques a partir de los cuales se determinan las paletas predictivas, en este ejemplo, las primeras paletas 184. Las paletas predictivas pueden asociarse a una o más CU vecinas de las CU actualmente codificadas (por ejemplo, tales como las CU vecinas (espacialmente o en función del orden de exploración) o las muestras más frecuentes de un vecino casual), es decir, la segunda CU 188. Las paletas de las una o más CU contiguas pueden estar asociadas a una paleta predictora. En algunos ejemplos, tales como el ejemplo ilustrado en la figura 4, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden ubicar una CU contigua izquierda, la primera CU 180, al determinar una paleta predictiva para la segunda CU 188. En otros ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden ubicar una o más CU en otras posiciones relativas a la segunda CU 188, tal como una CU superior, la CU 196.

[0145] El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar una CU para la predicción de la paleta basándose en una jerarquía. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden identificar inicialmente la CU contigua izquierda, la primera CU 180, para la predicción de paleta. Si la CU contigua izquierda no está disponible para la predicción (por ejemplo, la CU contigua izquierda está codificada con una modalidad diferente a la de codificación basada en paletas, tal como una modalidad de intrapredicción o una modalidad de intrapredicción, o está ubicada en el borde más a la izquierda de una imagen o segmento), el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden identificar la CU contigua superior, la CU 196. El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden continuar buscando una CU disponible de acuerdo a un orden predeterminado de ubicaciones hasta localizar una CU que tenga una paleta disponible para la predicción de paleta. En algunos ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar una paleta predictiva basándose en múltiples bloques y/o muestras reconstruidas de un bloque vecino.

[0146] Si bien el ejemplo de la figura 4 ilustra las primeras paletas 184 como paletas predictivas de una CU única, la primera CU 180, en otros ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden ubicar paletas para la predicción a partir de una combinación de las CU vecinas. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo pueden aplicar una o más fórmulas, funciones, reglas o similares para generar una paleta basándose en paletas entre una, o una combinación de una, pluralidad de las CU vecinas.

[0147] En otros ejemplos más, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden construir una lista de candidatos que incluya una serie de candidatos potenciales para la predicción de paleta. En tales ejemplos, el codificador de vídeo 20 puede codificar un índice para la lista de candidatos, para indicar la CU candidata en la lista de la cual se

selecciona la CU actual usada para la predicción de paleta (por ejemplo, copia la paleta). El decodificador de vídeo 30 puede construir la lista de candidatos de la misma manera, decodificar el índice y usar el índice decodificado para seleccionar la paleta de la CU correspondiente, a usar con la CU actual.

[0148] En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden construir una lista de candidatos que incluye una CU que está situada por encima de la CU que se está codificando actualmente y una CU que está situada a la izquierda de la CU que se está codificando actualmente. En este ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para indicar la selección de candidatos. Por ejemplo, el codificador de vídeo 20 puede codificar un indicador que tiene un valor de cero para indicar que la paleta para la CU actual se copia desde la CU situada a la izquierda de la CU actual. El codificador de vídeo 20 puede codificar un indicador que tiene un valor de cero para indicar que la paleta para la CU actual está copiada de la CU situada encima de la CU actual. El decodificador de vídeo 30 decodifica el indicador y selecciona la CU adecuada para la predicción de paleta.

[0149] En otros ejemplos más, el codificador de vídeo 20 y/o el decodificador de vídeo 30 determinan la paleta para la CU que se está codificando actualmente en función de la frecuencia con la que los valores de muestra incluidos en otras una o más paletas aparecen en una o más CU contiguas. Por ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden rastrear los colores asociados a los valores de índices utilizados con mayor frecuencia durante la codificación de un número predeterminado de las CU. El codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden incluir los colores más frecuentemente utilizados en la paleta para la CU que se está codificando actualmente.

[0150] Como se ha señalado anteriormente, en algunos ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo pueden copiar una paleta completa a partir de una CU vecina para codificar una CU actual. Adicional o alternativamente, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden realizar una predicción de paleta basada en las entradas. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para cada entrada de una paleta, indicando si las entradas respectivas se predicen basándose en una paleta predictiva (por ejemplo, una paleta de otra CU). En este ejemplo, el codificador de vídeo 20 puede codificar un indicador que tiene un valor igual a uno para una entrada dada cuando la entrada es un valor predicho a partir de una paleta predictiva (por ejemplo, una entrada correspondiente de una paleta asociada a una CU vecina). El codificador de vídeo 20 puede codificar un indicador que tiene un valor igual a cero para una entrada en particular, para indicar que la entrada en particular no está predicha a partir de una paleta de otra CU. En este ejemplo, el codificador de vídeo 20 también puede codificar datos adicionales que indiquen el valor de la entrada de paleta no predicha.

[0151] En el ejemplo de la FIG. 4, las segundas paletas 192 incluyen cuatro entradas 208-214 que tienen el valor de índice de entrada 1, el valor de índice de entrada 2, el valor de índice de entrada 3 y el índice de entrada 4, respectivamente. Las entradas 208-214 relacionan los valores de índice con los valores de píxel, incluyendo el valor de píxel A, el valor de píxel B, el valor de píxel C y el valor de píxel D, respectivamente. De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden usar cualquiera de las técnicas descritas anteriormente para ubicar la primera CU 180 con fines de predicción de paleta y copiar las entradas 1 a 3 de las primeras paletas 184 a las entradas 1 a 3 de las segundas paletas 192, para codificar la segunda CU 188. De este modo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar las segundas paletas 192 basándose en las primeras paletas 184. Además, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar datos para que la entrada 4 se incluya con las segundas paletas 192. Dicha información puede incluir el número de entradas de paleta no predichas a partir de una paleta predictora y los valores de píxeles correspondientes a esas entradas de paleta.

[0152] En algunos ejemplos, según aspectos de esta divulgación, uno o más elementos sintácticos pueden indicar si las paletas, tales como las segundas paletas 192, se predicen completamente a partir de una paleta predictiva (mostrada en la figura 4 como las primeras paletas 184, pero que pueden estar compuestas de entradas de uno o más bloques) o si se predicen entradas particulares de las segundas paletas 192. Por ejemplo, un elemento sintáctico inicial puede indicar si se predicen todas las entradas. Si el elemento sintáctico inicial indica que no se predicen todas las entradas (por ejemplo, un indicador que tiene un valor de 0), uno o más elementos sintácticos adicionales pueden indicar cuáles entradas de las segundas paletas 192 se predicen a partir de la paleta predictiva.

[0153] De acuerdo a algunos aspectos de esta divulgación, cierta información asociada a la predicción de paleta puede deducirse de una o más características de los datos que se codifican. Es decir, en lugar de que el codificador de vídeo 20 codifique elementos sintácticos (y que el decodificador de vídeo 30 decodifique dichos elementos sintácticos), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden realizar la predicción de paleta basándose en una o más características de los datos que se codifican.

[0154] En un ejemplo, con fines de ilustración, el valor del indicador_paleta_pred, descrito anteriormente, se puede deducir a partir de uno o más entre, como ejemplos, el tamaño de la CU que se está codificando, el tipo de trama, el espacio cromático, el componente cromático, el tamaño de trama, la velocidad de trama, el identificador de capa en la codificación de vídeo ajustable a escala o el identificador de vista en la codificación de múltiples vistas. Es decir, con respecto al tamaño de la CU como ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar que el indicador_paleta_pred descrito anteriormente es igual a uno para cualquier CU que supere un tamaño predeterminado. En este ejemplo, el indicador_paleta_pred no necesita ser señalizado en el flujo de bits codificado.

[0155] Si bien se ha descrito anteriormente con respecto al indicador_paleta_pred, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden, también o alternativamente, deducir otra información asociada a la predicción de paleta, tal como la CU candidata a partir de la cual se utiliza la paleta para la predicción, o las reglas para construir la candidatos de predicción de paleta, basándose en una o más características de los datos que están siendo codificados.

[0156] De acuerdo a otros aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden construir una paleta sobre la marcha. Por ejemplo, cuando se codifica inicialmente la segunda CU 188, no hay ninguna entrada en las paletas 192. A medida que el codificador de vídeo 20 y el decodificador de vídeo 30 codifican valores nuevos para los píxeles de la segunda CU 188, cada nuevo valor se incluye en las paletas 192. Es decir, por ejemplo, el codificador de vídeo 20 agrega valores de píxel a las paletas 192 a medida que los valores de píxel se generan y señalizan para las posiciones en la CU 188. Como el codificador de vídeo 20 codifica píxeles relativamente más tarde en la CU, el codificador de vídeo 20 puede codificar píxeles que tengan los mismos valores que los que ya están incluidos en la paleta, utilizando valores de índices en lugar de señalar los valores de píxeles. De manera similar, cuando el decodificador de vídeo 30 recibe un nuevo valor de píxel (por ejemplo, señalado por el codificador de vídeo 20) para una posición en la segunda CU 188, el decodificador de vídeo 30 incluye el valor de píxel en las paletas 192. Cuando las posiciones de píxeles, decodificadas relativamente más tarde en la segunda CU 188, tienen valores de píxeles que se han agregado a las segundas paletas 192, el decodificador de vídeo 30 puede recibir información tal como, por ejemplo, valores de índices, que identifican los correspondiente valores de píxeles en las segundas paletas 192, para la reconstrucción de los valores de píxeles de la segunda CU 188.

[0157] En algunos ejemplos, como se describe con mayor detalle a continuación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden mantener las paletas 184 y 192 en o por debajo de un tamaño de paleta máximo. De acuerdo a aspectos de esta divulgación, si se alcanza un tamaño máximo de paleta, por ejemplo, según las segundas paletas 192 se construyen dinámicamente sobre la marcha, entonces el codificador de vídeo 20 y/o el decodificador de vídeo 30 realizan el mismo proceso para eliminar una entrada de las segundas paletas 192. Un proceso ejemplar para eliminar entradas de paleta es una técnica de 'primero en entrar, primero en salir' (FIFO), en la que el codificador de vídeo 20 y el decodificador de vídeo 30 eliminan la entrada más antigua de una paleta. En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden eliminar la entrada de paleta menos frecuentemente utilizada de la paleta. En otro ejemplo más, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden ponderar tanto los procesos FIFO como los procesos de frecuencia de uso para determinar qué entrada eliminar. Es decir, la eliminación de una entrada puede basarse en la antigüedad de la entrada y la frecuencia con la que se utiliza.

[0158] De acuerdo a algunos aspectos, si se elimina una entrada (valor de píxel) de una paleta y el valor de píxel vuelve a aparecer en una posición posterior en la CU que se está codificando, el codificador de vídeo 20 puede codificar el valor de píxel en lugar de incluir una entrada en la paleta y codificar un índice. Adicional o alternativamente, el codificador de vídeo 20 puede reingresar entradas de paleta en la paleta después de haber sido eliminadas, por ejemplo, a medida que el codificador de vídeo 20 y el decodificador de vídeo 30 recorren las posiciones en la CU.

[0159] En algunos ejemplos, las técnicas para obtener una paleta sobre la marcha se pueden combinar con una o más técnicas diferentes para determinar una paleta. En particular, como ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar inicialmente las segundas paletas 192 (por ejemplo, usando la predicción de paleta para predecir las segundas paletas 192 a partir de las primeras paletas 184) y pueden actualizar las segundas paletas 192 cuando se codifican los píxeles de la segunda CU 188. Por ejemplo, al transmitir la paleta inicial, el codificador de vídeo 20 puede agregar valores a la paleta inicial o cambiar los valores en la paleta inicial a medida que se recorren los valores de píxeles de ubicaciones adicionales en la CU. Del mismo modo, al recibir una paleta inicial, el decodificador de vídeo 30 puede agregar valores a la paleta inicial o cambiar los valores en la paleta inicial a medida que se recorren los valores de píxeles de ubicaciones adicionales en la CU.

[0160] El codificador de vídeo 20 puede, en algunos ejemplos, señalar si la CU actual utiliza la transmisión de una paleta completa, o la generación de paletas sobre la marcha, o una combinación de transmisión de una paleta inicial con la actualización de la paleta inicial mediante obtención sobre la marcha. En algunos ejemplos, la paleta inicial puede ser una paleta completa en el tamaño máximo de paleta, en cuyo caso se pueden cambiar los valores en la paleta inicial. En otros ejemplos, la paleta inicial puede ser más pequeña que el tamaño máximo de paleta, en cuyo caso el codificador de vídeo 20 y el decodificador de vídeo 30 pueden agregar valores y/o cambiar valores de la paleta inicial.

[0161] De acuerdo a aspectos de esta divulgación, el tamaño de las paletas, tales como las primeras paletas 184 y las segundas paletas 192, por ejemplo, en términos de la cantidad de valores de píxeles que se incluyen en la paleta, puede ser fijo o puede señalizarse utilizando uno o más elementos sintácticos en un flujo de bits codificado. Por ejemplo, según algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar códigos unarios o códigos unarios truncados (por ejemplo, códigos que se truncan en un límite máximo del tamaño de la paleta) para codificar el tamaño de la paleta. De acuerdo a otros aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar códigos Exponencial-Golomb o Rice-Golomb para codificar el tamaño de la paleta.

[0162] De acuerdo a otros aspectos más, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos que indiquen el tamaño de la paleta después de cada entrada de la paleta. Con respecto a las segundas paletas 192, como ejemplo, el codificador de vídeo 20 puede codificar un indicador de parada después de cada una de las entradas 208 a 214. En este ejemplo, un indicador de parada igual a uno puede especificar que la entrada que se está codificando actualmente es la entrada final de las segundas paletas 192, mientras que un indicador de parada igual a cero puede indicar que hay entradas adicionales en las segundas paletas 192. Por consiguiente, el codificador de vídeo 20 puede codificar indicadores de parada que tengan un valor de cero después de cada una de las entradas 208 a 212 y un indicador de parada que tenga un valor de uno después de la entrada 214. En algunos casos, es posible que el indicador de parada no se incluya en el flujo de bits cuando la paleta construida alcance un límite máximo de tamaño de paleta. Si bien los ejemplos anteriores divulgan técnicas para señalar explícitamente el tamaño de las paletas, en otros ejemplos, el tamaño de las paletas también se puede transmitir o deducir condicionalmente basándose en la llamada información lateral (por ejemplo, información característica, tal como el tamaño de la CU que se está codificando, el tipo de trama, el espacio cromático, el componente cromático, el tamaño de trama, la velocidad de trama, el identificador de capa en la codificación de vídeo ajustable a escala o el identificador de vista en la codificación de múltiples vistas, como se ha señalado anteriormente).

[0163] Las técnicas de esta divulgación incluyen la codificación de datos sin pérdidas o, alternativamente, con algunas pérdidas (codificación con pérdidas). Por ejemplo, con respecto a la codificación con pérdidas, el codificador de vídeo 20 puede codificar los píxeles de una CU sin aparear exactamente los valores de píxeles de las paletas con los valores de píxeles reales en la CU. Cuando las técnicas de esta divulgación se aplican a la codificación con pérdidas, se pueden aplicar algunas restricciones a la paleta. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden cuantizar paletas, tales como las primeras paletas 184 y las segundas paletas 192. Es decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden combinar (cuantizar) entradas de una paleta cuando los valores de píxeles de las entradas están dentro de una distancia predeterminada entre sí. En otras palabras, si ya hay un valor de paleta que está dentro de un margen de error de un nuevo valor de paleta, el nuevo valor de paleta no se agrega a la paleta. En otro ejemplo, una pluralidad de valores de píxeles diferentes en un bloque puede correlacionarse con una sola entrada de paleta o, de manera equivalente, con un solo valor de píxel de paleta.

[0164] El decodificador de vídeo 30 puede decodificar valores de píxel de la misma manera, independientemente de si una paleta particular no tiene pérdidas o tiene pérdidas. Como ejemplo, el decodificador de vídeo 30 puede usar un valor de índice transmitido por el codificador de vídeo 20 para una posición de píxel dada en un bloque codificado, para seleccionar una entrada en la paleta para la posición de píxel, sin que importe si la paleta no tiene pérdidas o tiene pérdidas. En este ejemplo, el valor de píxel de la entrada de paleta se utiliza como el valor de píxel en el bloque codificado, ya sea que coincida o no exactamente con el valor de píxel original.

[0165] En un ejemplo de codificación con pérdidas, con fines de ilustración, el codificador de vídeo 20 puede determinar una cota de error, denominada valor delta. Una entrada candidata de valor de píxel Plt_cand puede corresponder a un valor de píxel en una posición en un bloque a codificar, tal como CU o PU. Durante la construcción de la paleta, el codificador de vídeo 20 determina la diferencia absoluta entre la entrada candidata de valor de píxel Plt_cand y todas las entradas de valores de píxeles existentes en la paleta. Si todas las diferencias absolutas entre la entrada de valor de píxel candidato Plt_cand y las entradas de valores de píxel existentes en la paleta son mayores que el valor delta, el codificador de vídeo 20 puede agregar el valor de píxel candidato a la paleta como una entrada. Si una diferencia absoluta entre la

entrada de valor de píxel Plt_cand y al menos una entrada de valor de píxel existente en la paleta es igual o menor que el valor delta, el codificador de vídeo 20 puede no agregar la entrada de valor de píxel candidato Plt_cand a la paleta. Por lo tanto, cuando se codifica la entrada de valor de píxel Plt_cand, el codificador de vídeo 20 puede seleccionar la entrada con el valor de píxel que sea el más cercano a la entrada de valor de píxel Plt_cand, introduciendo así alguna pérdida en el sistema. Cuando una paleta consta de múltiples componentes (por ejemplo, tres componentes cromáticos), la suma de la diferencia absoluta de los valores de componentes individuales se puede utilizar para comparar con el valor delta. De forma alternativa o adicional, la diferencia absoluta para cada valor de componente puede compararse con un segundo valor delta.

[0166] En algunos ejemplos, las entradas de valores de píxel existentes en la paleta indicada anteriormente pueden haberse agregado utilizando un proceso similar de comparación con delta. En otros ejemplos, los valores de píxeles existentes en la paleta se pueden haber agregado usando otros procesos. Por ejemplo, se pueden agregar una o más entradas de valores iniciales de píxel a una paleta (sin una comparación con delta) para iniciar el proceso de comparación con delta de la construcción de la paleta. El proceso descrito anteriormente puede implementarse mediante el codificador de vídeo 20 y/o el decodificador de vídeo 30 para producir paletas de luminancia y/o croma.

[0167] Las técnicas descritas anteriormente con respecto a la construcción de la paleta también pueden ser usadas por el codificador de vídeo 20 y el decodificador de vídeo 30 durante la codificación de píxeles. Por ejemplo, cuando se codifica un valor de píxel, el codificador de vídeo 20 puede comparar el valor del píxel con los valores de píxel de las entradas en la paleta. Si la diferencia absoluta de valores de píxel entre el valor del píxel y una de las entradas en la paleta es igual o menor que un valor delta, el codificador de vídeo 20 puede codificar el valor del píxel como la entrada de la paleta. Es decir, en este ejemplo, el codificador de vídeo 20 codifica el valor de píxel utilizando una de las entradas de la paleta cuando el valor de píxel produce una diferencia absoluta suficientemente pequeña (por ejemplo, dentro de un rango predeterminado) en comparación con la entrada de la paleta.

[0168] En algunos ejemplos, el codificador de vídeo 20 puede seleccionar la entrada de la paleta que produce la diferencia absoluta de valor de píxel más pequeña (en comparación con el valor de píxel que se está codificando) para codificar el valor de píxel. Como ejemplo, el codificador de vídeo 20 puede codificar un índice para indicar una entrada de paleta que se usará para el valor de píxel, por ejemplo, la entrada de valor de píxel de paleta que se usará para reconstruir el valor de píxel codificado en el decodificador de vídeo 30. Si la diferencia absoluta de valor de píxel entre el valor del píxel y todas las entradas en la paleta es mayor que delta, el codificador no puede usar una de las entradas de la paleta para codificar el valor del píxel, y en cambio puede transmitir el valor de píxel del píxel a codificar al decodificador de vídeo 30 (y posiblemente agregar el valor de píxel como una entrada a la paleta).

[0169] En otro ejemplo, el codificador de vídeo 20 puede seleccionar una entrada de una paleta para codificar un valor de píxel. El codificador de vídeo 20 puede usar la entrada seleccionada como un valor de píxel predictivo. Es decir, el codificador de vídeo 20 puede determinar un valor residual que representa una diferencia entre el valor de píxel real y la entrada seleccionada y codificar el residuo. El codificador de vídeo 20 puede generar valores residuales para píxeles en un bloque que son predichos por las entradas de una paleta, y puede generar un bloque de residuos que incluya respectivos valores de píxeles residuales para el bloque de píxeles. El codificador de vídeo 20 puede posteriormente aplicar transformación y cuantización (como se ha indicado anteriormente con respecto a la figura 2) al bloque de residuos. De esta manera, el codificador de vídeo 20 puede generar coeficientes de transformación residual cuantizados.

[0170] El decodificador de vídeo 30 puede transformar inversamente y cuantizar inversamente los coeficientes de transformación para reproducir el bloque residual. El decodificador de vídeo 30 puede luego reconstruir un valor de píxel utilizando el valor de entrada de la paleta predictiva y el valor residual para el valor de píxel. Por ejemplo, el decodificador de vídeo 30 puede combinar el valor residual con el valor de entrada de paleta para reconstruir el valor de píxel codificado.

[0171] En algunos ejemplos, el valor delta puede ser diferente para diferentes tamaños de CU, tamaños de imagen, espacios de color o componentes de color diferentes. El valor delta puede determinarse o determinarse basándose en diversas condiciones de codificación. Por ejemplo, el codificador de vídeo 20 puede señalar el valor delta al decodificador de vídeo 30 usando una sintaxis de alto nivel, tal como la sintaxis en PPS, SPS, VPS y/o encabezado de fragmento. En otros ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden preconfigurarse para utilizar el mismo valor delta fijo. En otros ejemplos más, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden obtener adaptativamente el valor delta basándose en información lateral (por ejemplo, tal como el tamaño de CU, el espacio cromático, el componente cromático o similares, como se ha indicado anteriormente).

[0172] En algunos ejemplos, se puede incluir un modo de paleta de codificación con pérdidas como un modo de codificación HEVC. Por ejemplo, las modalidades de codificación pueden incluir una modalidad de intrapredicción, una modalidad de interpredicción, una modalidad de paleta de codificación sin pérdidas y una modalidad de paleta de codificación con pérdidas. En la codificación HEVC, como se ha indicado anteriormente con respecto a las figuras 2 y 3, se usa un parámetro de cuantización (QP) para controlar la distorsión admitida. El valor de delta para la codificación basada en paleta puede calcularse o determinarse de otra manera como una función del QP. Por ejemplo, el valor delta descrito anteriormente puede ser $1 \ll (QP/6)$ o $1 \ll ((QP + d)/6)$ donde d es una constante y " $\ll$ " representa el operador de desplazamiento a la izquierda bit a bit.

[0173] La generación de una paleta utilizando las técnicas de codificación con pérdidas descritas en esta divulgación puede realizarse mediante el codificador de vídeo 20, el decodificador de vídeo 30 o ambos. Por ejemplo, el codificador de vídeo 20 puede generar entradas en una paleta para una CU utilizando las técnicas de comparación con delta descritas anteriormente y señalar información para la construcción de la paleta, para su uso por el decodificador de vídeo 30. Es decir, el codificador de vídeo 20 puede configurarse para señalar información que indique valores de píxel para las entradas en una paleta para una CU, y luego codificar valores de píxel utilizando los valores de píxel asociados a dichas entradas de paleta. El decodificador de vídeo 30 puede construir una paleta usando dicha información y luego usar las entradas para decodificar los valores de píxeles de un bloque codificado. En algunos ejemplos, el codificador de vídeo 20 puede señalar valores de índices que identifican entradas de paleta para una o más posiciones de píxeles del bloque codificado, y el decodificador de vídeo 30 puede usar los valores de índices para recuperar las entradas pertinentes de valores de píxeles de la paleta.

[0174] En otros ejemplos, el decodificador de vídeo 30 se puede configurar para construir una paleta aplicando las técnicas de comparación con delta descritas anteriormente. Por ejemplo, el decodificador de vídeo 30 puede recibir valores de píxel para posiciones dentro de un bloque codificado y determinar si las diferencias absolutas entre los valores de píxel y las entradas de valores de píxel existentes en la paleta son mayores que un valor delta. Si es así, el decodificador de vídeo 30 puede agregar los valores de píxel como entradas en la paleta, por ejemplo, para su uso posterior en la decodificación basada en paleta de los valores de píxel para otras posiciones de píxel del bloque, usando los correspondientes valores de índices señalizados por el codificador de vídeo 20. En este caso, el codificador de vídeo 20 y el decodificador de vídeo 30 aplican los mismos procesos o procesos similares para generar la paleta. Si no, el decodificador de vídeo 30 puede no agregar los valores de píxeles a la paleta.

[0175] En un ejemplo con fines de ilustración, el decodificador de vídeo 30 puede recibir valores de índices o valores de píxeles para varias posiciones de píxeles en un bloque. Si se recibe un valor de índice para una posición de píxel, por ejemplo, el decodificador de vídeo 30 puede usar el valor de índice para identificar una entrada en la paleta y usar el valor de píxel de la entrada de paleta para la posición de píxel. Si se recibe un valor de píxel para la posición de píxel, el decodificador de vídeo 30 puede usar el valor de píxel recibido para la posición de píxel, y también aplicar la comparación con delta para determinar si el valor de píxel debería agregarse a la paleta y luego usarse para la codificación de paleta.

[0176] En el lado del codificador, si un valor de píxel para una posición en un bloque produce una diferencia absoluta entre el valor de píxel y una entrada de valor de píxel existente en la paleta que es menor o igual que el valor delta, el codificador de vídeo 20 puede enviar un valor de índice para identificar la entrada en la paleta para su uso en la reconstrucción del valor de píxel para esa posición. Si un valor de píxel para una posición en un bloque produce valores de diferencia absoluta entre el valor de píxel y las entradas de valores de píxel existentes en la paleta que son todas mayores que el valor delta, el codificador de vídeo 20 puede enviar el valor de píxel y agregar el valor de píxel como una nueva entrada en la paleta. Para construir la paleta, el decodificador de vídeo 30 puede usar valores delta señalizados por el codificador, basarse en un valor delta fijo o conocido, o deducir u obtener un valor delta, por ejemplo, como se ha descrito anteriormente.

[0177] Como se ha indicado anteriormente, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden usar modalidades de codificación que incluyen una modalidad de intrapredicción, una modalidad de interpredicción, una modalidad de paleta de codificación sin pérdidas y una modalidad de paleta de codificación con pérdidas cuando se codifican datos de vídeo. De acuerdo a algunos aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar uno o más elementos sintácticos que indican si la codificación basada en paletas está habilitada. Por ejemplo, en cada CU, el codificador de vídeo 20 puede codificar un elemento sintáctico, tal como un indicador_Modalidad_PLT. El indicador_Modalidad_PLT u otro elemento sintáctico puede indicar si se ha de usar una modalidad de codificación basada en paleta para una CU determinada (o una PU en otros ejemplos). Por ejemplo, este

indicador puede ser señalizado en un flujo de bits de vídeo codificado a nivel de CU, y luego recibido por el decodificador de vídeo 30 al decodificar el flujo de bits de vídeo codificado.

[0178] En este ejemplo, un valor de este indicador_Modalidad_PLT igual a 1 puede especificar que la CU actual se codifica utilizando una modalidad de codificación basada en paletas. En este caso, el decodificador de vídeo 30 puede aplicar la modalidad de codificación basada en paletas para decodificar la CU. En algunos ejemplos, un elemento sintáctico puede indicar uno entre una pluralidad de diferentes modalidades de codificación basadas en paletas para la CU (por ejemplo, con pérdidas o sin pérdidas). Un valor de este indicador_Modalidad_PLT igual a 0 puede especificar que la CU actual se codifica utilizando una modalidad diferente a la modalidad de paleta. Por ejemplo, se puede usar cualquiera entre varias modalidades de codificación interpredictiva, intrapredictiva u otras. Cuando un valor del indicador_Modalidad_PLT es 0, el codificador de vídeo 20 también puede codificar datos adicionales para indicar la modalidad específica utilizada para codificar la CU respectiva (por ejemplo, una modalidad de codificación HEVC). El uso del indicador_Modalidad_PLT se describe a modo de ejemplo. Sin embargo, en otros ejemplos, otros elementos sintácticos, tales como los códigos de múltiples bits, se pueden usar para indicar si la modalidad de codificación basada en paletas se ha de usar para una CU (o PU en otros ejemplos) o para indicar cuáles, entre una pluralidad de modalidades, deben ser utilizadas para la codificación.

[0179] En algunos ejemplos, el indicador descrito anteriormente u otros elementos sintácticos pueden transmitirse a un nivel más alto que el nivel de CU (o PU). Por ejemplo, el codificador de vídeo 20 puede señalar un indicador de ese tipo a un nivel de fragmento. En este caso, un valor igual a 1 indica que todas las CU en el segmento se codifican utilizando la modalidad de paleta. En este ejemplo, ninguna información adicional de modalidad, por ejemplo, para la modalidad de paleta u otras modalidades, se señala en el nivel de CU. En otro ejemplo, el codificador de vídeo 20 puede señalar un indicador de este tipo en un PPS, SPS o VPS.

[0180] De acuerdo a algunos aspectos de esta divulgación, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden codificar uno o más elementos sintácticos (por ejemplo, tales como el indicador descrito anteriormente) en uno de los niveles de fragmento, PPS, SPS o VPS, especificando si la modalidad de paleta está habilitada o inhabilitada para el fragmento, imagen, secuencia o similar en particular, mientras que el indicador_Modalidad_PLT indica si la modalidad de codificación basada en paletas se usa para cada CU. En este caso, si un indicador u otro elemento sintáctico enviado a nivel de fragmento, PPS, SPS o VPS indica que la modalidad de codificación de paleta está inhabilitada, en algunos ejemplos, puede que no sea necesario señalar el indicador_Modalidad_PLT para cada CU. Alternativamente, si un indicador u otro elemento sintáctico enviado a nivel de fragmento, PPS, SPS o VPS indica que la modalidad de codificación de paleta está habilitada, el indicador_Modalidad_PLT puede señalizarse adicionalmente para indicar si se ha de usar la modalidad de codificación basada en paletas para cada CU. De nuevo, como se ha mencionado anteriormente, la aplicación de estas técnicas para indicar la codificación basada en paletas de una CU podría usarse, adicional o alternativamente, para indicar la codificación basada en paletas de una PU.

[0181] En algunos ejemplos, los elementos sintácticos descritos anteriormente pueden señalizarse condicionalmente en el flujo de bits. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 solo pueden codificar o decodificar, respectivamente, los elementos sintácticos basándose en el tamaño de la CU, el tipo de trama, el espacio cromático, el componente cromático, el tamaño de trama, la velocidad de trama, el Identificador de capa en la codificación de vídeo ajustable a escala o el Identificador de vista en la codificación de múltiples vistas.

[0182] Si bien los ejemplos descritos anteriormente se refieren a la señalización explícita, por ejemplo, con uno o más elementos sintácticos en un flujo de bits, en otros ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden determinar implícitamente si una modalidad de codificación de paleta está activa y/o se usa para codificar un bloque particular. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar si la codificación basada en paletas se usa para un bloque basándose en, por ejemplo, el tamaño de la CU, el tipo de trama, el espacio cromático, el componente cromático, el tamaño de trama, la velocidad de trama, el identificador de capa en la codificación de vídeo ajustable a escala o el identificador de vista en la codificación de múltiples vistas.

[0183] Si bien las técnicas de la figura 4 se describen anteriormente en el contexto de las CU (HEVC), debería entenderse que las técnicas también pueden aplicarse a unidades de predicción (PU) o en otros procesos y/o normas de codificación de vídeo.

[0184] La figura 5 es un diagrama conceptual que ilustra un ejemplo de determinación de índices para una paleta para un bloque de píxeles, congruente con las técnicas de esta divulgación. Por ejemplo, la figura 5 incluye una correlación 240

de valores de índices (valores 1, 2 y 3) que relacionan posiciones respectivas de píxeles asociados a los valores de índices con una entrada de paletas 244. Las paletas 244 se pueden determinar de manera similar a las primeras paletas 184 y las segundas paletas 192 descritas anteriormente con respecto a la figura 4.

[0185] De nuevo, las técnicas de la figura 5 se describen en el contexto del codificador de vídeo 20 (figura 1 y figura 2) y del decodificador de vídeo 30 (figura 1 y figura 3) y con respecto a la norma de codificación de vídeo HEVC, con fines de explicación. Sin embargo, debería entenderse que las técnicas de esta divulgación no están limitadas de esta manera, y que pueden ser aplicadas por otros procesadores y/o dispositivos de codificación de vídeo en otros procesos y/o normas de codificación de vídeo.

[0186] Si bien la correlación 240 se ilustra en el ejemplo de la figura 5 como incluyente de un valor de índice para cada posición de píxel, debería entenderse que, en otros ejemplos, no todas las posiciones de píxel pueden asociarse a un valor de índice que relacione el valor de píxel con una entrada de las paletas 244. Es decir, como se ha indicado anteriormente, en algunos ejemplos, el codificador de vídeo 20 puede codificar (y el decodificador de vídeo 30 puede obtener, a partir de un flujo de bits codificado) una indicación de un valor de píxel real (o su versión cuantizada) para una posición en la correlación 240 si el valor de píxel no está incluido en las paletas 244.

[0187] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar un mapa adicional que indique qué posiciones de píxel están asociadas a valores de índice. Por ejemplo, supongamos que la entrada (i, j) en el mapa corresponde a la posición (i, j) de una CU. El codificador de vídeo 20 puede codificar uno o más elementos sintácticos para cada entrada de la correlación (es decir, cada posición de píxel) indicando si la entrada tiene un valor de índice asociado. Por ejemplo, el codificador de vídeo 20 puede codificar un indicador que tiene un valor de uno para indicar que el valor de píxel en la ubicación (i, j) en la CU es uno de los valores en las paletas 244. El codificador de vídeo 20 puede, en dicho ejemplo, codificar también un índice de paleta (mostrado en el ejemplo de la FIG. 5 como los valores 1 a 3) para indicar ese valor de píxel en la paleta y para permitir que el decodificador de vídeo reconstruya el valor de píxel. En casos en los que las paletas 244 incluyen una sola entrada y un valor de píxel asociado, el codificador de vídeo 20 puede omitir la señalización del valor de índice. El codificador de vídeo 20 puede codificar que el indicador tenga un valor de cero para indicar que el valor de píxel en la ubicación (i, j) en la CU no es uno de los valores en las paletas 244. En este ejemplo, el codificador de vídeo 20 también puede codificar una indicación del valor de píxel para su uso por el decodificador de vídeo 30 al reconstruir el valor de píxel. En algunos casos, el valor de píxel puede codificarse de una manera que incluye pérdidas.

[0188] El valor de un píxel en una posición de una CU puede proporcionar una indicación de valores de otros uno o más píxeles en otras posiciones de la CU. Por ejemplo, puede haber una probabilidad relativamente alta de que las posiciones de píxeles adyacentes de una CU tengan el mismo valor de píxel o se puedan correlacionar con el mismo valor de índice (en el caso de codificación con pérdidas, en la que se puede correlacionar más de un píxel con un único valor de índice).

[0189] En consecuencia, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican una serie de píxeles o valores de índices consecutivos, en un orden de exploración dado, que tienen el mismo valor de píxel o valor de índice. Como se ha indicado anteriormente, la cadena de valores de índice o píxel de valor similar se puede denominar en el presente documento una serie. En un ejemplo con fines de ilustración, si dos píxeles o índices consecutivos en un orden de exploración dado tienen valores diferentes, la racha es igual a cero. Si dos píxeles o índices consecutivos en un orden de exploración dado tienen el mismo valor, pero el tercer píxel o índice en el orden de exploración tiene un valor diferente, la racha es igual a uno. Para tres índices o píxeles consecutivos con el mismo valor, la racha es dos, y así sucesivamente. El decodificador de vídeo 30 puede obtener los elementos sintácticos que indican una racha a partir de un flujo de bits codificado y utilizar los datos para determinar el número de ubicaciones consecutivas que tienen el mismo valor de índice o píxel.

[0190] En un ejemplo con fines de ilustración, considérese la línea 248 de la correlación 240. Suponiendo una dirección de exploración horizontal, de izquierda a derecha, la línea 248 incluye cinco valores de índice de "2" y tres valores de índice de "3". De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede codificar un valor de índice de 2 para la primera posición de la línea 248 en la dirección de exploración. Además, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican la racha de valores consecutivos en la dirección de exploración que tienen el mismo valor de índice que el valor de índice señalado. En el ejemplo de la línea 248, el codificador de vídeo 20 puede señalar una racha de 4, indicando así que los valores de índice de las siguientes cuatro posiciones en la dirección de exploración comparten el mismo valor de índice que el valor de índice señalado. El codificador de vídeo 20 puede realizar el mismo proceso para el siguiente valor de índice diferente en la línea 248. Es decir, el codificador de vídeo 20

puede codificar un valor de índice de 3 y uno o más elementos sintácticos que indican una serie de dos. El decodificador de vídeo 30 puede obtener los elementos sintácticos que indican el valor del índice y el número de índices consecutivos en la dirección de exploración que tienen el mismo valor de índice (la racha).

[0191] Como se ha señalado anteriormente, los índices de una correlación se recorren en un orden particular. De acuerdo a aspectos de esta divulgación, la dirección de exploración puede ser vertical, horizontal o en diagonal (por ejemplo, 45 grados o 135 grados diagonalmente en bloque). En algunos ejemplos, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para cada bloque que indica una dirección de exploración para recorrer los índices del bloque. Adicional o alternativamente, la dirección de exploración puede ser señalizada o deducida basándose en la llamada información lateral, tal como, por ejemplo, tamaño de bloque, espacio cromático y/o componente cromático. El codificador de vídeo 20 puede especificar exploraciones para cada componente cromático de un bloque. Alternativamente, una exploración específica puede aplicarse a todos los componentes cromáticos de un bloque.

[0192] Por ejemplo, con respecto a una exploración basada en columnas, considérese la columna 252 de la correlación 240. Suponiendo una dirección de exploración vertical, de arriba a abajo, la columna 252 incluye seis valores de índice de "2" y dos valores de índice de "3". De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede codificar un valor de índice de 2 para la primera posición de la línea 252 en la dirección de exploración (en la parte superior relativa de la columna 252). Además, el codificador de vídeo 20 puede señalizar una racha de 5, indicando así que los valores de índices de las siguientes cinco posiciones en la dirección de exploración comparten el mismo valor de índice que el valor de índice señalizado. El codificador de vídeo 20 también puede codificar un valor de índice de 3 para la siguiente posición en la dirección de exploración y uno o más elementos sintácticos que indican una racha de uno. El decodificador de vídeo 30 puede obtener los elementos sintácticos que indican el valor del índice y el número de índices consecutivos en la dirección de exploración que tienen el mismo valor de índice (la racha).

[0193] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden realizar, adicional o alternativamente, la copia por líneas para una o más entradas de la correlación 240. La copia por líneas puede depender, en algunos ejemplos, de la dirección de exploración. Por ejemplo, el codificador de vídeo 20 puede indicar que un píxel o valor de índice para una entrada particular en una correlación es igual a un píxel o valor de índice en una línea encima de la entrada particular (para un recorrido horizontal) o en la columna a la izquierda de la entrada particular (para un recorrido vertical). El codificador de vídeo 20 también puede indicar, como una racha, el número de valores de píxeles o índices en el orden de exploración que son iguales a la entrada en la línea superior o la columna a la izquierda de la entrada particular. En este ejemplo, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden copiar los valores de píxel o índice a partir de la línea adyacente especificada y del número especificado de entradas para la línea de la correlación que se está codificando actualmente.

[0194] En un ejemplo con fines de ilustración, considérense las columnas 256 y 260 de la correlación 240. Suponiendo una dirección de exploración horizontal, de arriba hacia abajo, la columna 256 incluye tres valores de índice de "1", tres valores de índice de "2" y dos valores de índice de "3". La columna 260 incluye los mismos valores de índices que tienen el mismo orden en la dirección de exploración. De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos para la columna 260, lo que indica que toda la columna 260 se copia a partir de la columna 256. El uno o más elementos sintácticos pueden estar asociados con una primera entrada de la columna 260 en la parte superior relativa del mapa 240. El decodificador de vídeo 30 puede obtener los elementos sintácticos que indican la copia por líneas y copiar los valores de índices de la columna 256 para la columna 260 cuando se decodifica la columna 260.

[0195] De acuerdo a aspectos de esta divulgación, las técnicas para codificar las denominadas rachas de entradas pueden usarse conjuntamente con las técnicas de copiado por líneas, descritas anteriormente. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos (por ejemplo, un indicador) que indica si el valor de una entrada en una correlación se obtiene de una paleta o si el valor de una entrada en la correlación se obtiene de una línea codificada previamente en la correlación 240. El codificador de vídeo 20 también puede codificar uno o más elementos sintácticos que indican un valor de índice de una paleta o la ubicación de la entrada en la línea (la fila o columna). El codificador de vídeo 20 también puede codificar uno o más elementos sintácticos que indican un número de entradas consecutivas que comparten el mismo valor. El decodificador de vídeo 30 puede obtener dicha información a partir de un flujo de bits codificado y usar la información para reconstruir la correlación y los valores de píxeles para un bloque.

[0196] En un ejemplo con fines de ilustración, considérense las filas 264 y 268 del mapa 240. Suponiendo una dirección de escaneo horizontal, de izquierda a derecha, la fila 264 incluye tres valores de índice de "1", dos valores de índice de

"2" y tres valores de índice de "3". La fila 268 incluye cinco valores de índice de "1" y tres valores de índice de "3". En este ejemplo, el codificador de vídeo 20 puede identificar entradas particulares de la fila 264 seguidas por una racha cuando se codifican datos para la fila 268. Por ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican que la primera posición de la fila 268 (la posición más a la izquierda de la fila 268) es la misma que la primera posición de la fila 264. El codificador de vídeo 20 también puede codificar uno o más elementos sintácticos que indican que la siguiente racha de dos entradas consecutivas en la dirección de exploración en la fila 268 es la misma que la primera posición de la fila 264.

[0197] En algunos ejemplos, el codificador de vídeo 20 también puede determinar si codifica el píxel o valor de índice actual con respecto a una posición en otra fila (o columna) o codifica el píxel o valor de índice actual usando un elemento sintáctico de racha. Por ejemplo, después de codificar los uno o más elementos sintácticos que indican la primera posición de la fila 264 y la racha de dos entradas (indicadas arriba), el codificador de vídeo 20 puede codificar, para las posiciones cuarta y quinta en la línea 268 (de izquierda a derecha), uno o más elementos sintácticos que indican un valor de 1 para la cuarta posición y uno o más elementos sintácticos que indican una racha de 1. Por lo tanto, el codificador de vídeo 20 codifica estas dos posiciones sin referencia a otra línea (o columna). El codificador de vídeo 20 puede entonces codificar la primera posición con un valor de índice de 3 en la fila 268 con respecto a la fila superior 264 (por ejemplo, lo que indica una copia de la fila superior 264 y la racha de posiciones consecutivas en el orden de exploración que tienen el mismo valor de índice). En consecuencia, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 puede seleccionar entre codificar valores de píxel o de índice de una línea (o columna) con respecto a otros valores de la línea (o columna), por ejemplo, utilizando una racha, codificar valores de píxel o de índice de una línea (o columna) con respecto a valores de otra línea (o columna) o una combinación de lo que antecede. El codificador de vídeo 20 puede, en algunos ejemplos, efectuar una optimización de velocidad/distorsión para realizar la selección.

[0198] El decodificador de vídeo 30 puede recibir los elementos sintácticos descritos anteriormente y puede reconstruir la fila 268. Por ejemplo, el decodificador de vídeo 30 puede obtener datos que indiquen una ubicación particular en una fila vecina desde la cual copiar el valor de índice asociado para la posición de la correlación 240 que se está codificando actualmente. El decodificador de vídeo 30 también puede obtener datos que indican el número de posiciones consecutivas en el orden de exploración que tienen el mismo valor de índice.

[0199] En algunos casos, la línea desde la cual se copian las entradas puede estar directamente adyacente a la entrada de la línea que se está codificando actualmente (como se ilustra en los ejemplos de la Figura 5). Sin embargo, en otros ejemplos, el codificador de vídeo 20 y/o el decodificador de vídeo pueden almacenar temporalmente varias líneas, de manera que cualesquiera entre la serie de líneas de la correlación se puedan usar como entradas predictivas para una línea de la correlación que se está codificando actualmente. Por lo tanto, en algunos ejemplos, el valor de píxel para una entrada se puede señalar para que sea igual a un valor de píxel de una entrada en una fila inmediatamente arriba (o una columna a la izquierda), o dos o más filas arriba (o columna a la izquierda), de la fila actual.

[0200] En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y/o el descodificador de vídeo 30 pueden configurarse para almacenar las n filas de entradas anteriores antes de codificar una fila actual de entradas. En este ejemplo, el codificador de vídeo 20 puede indicar la fila predictiva (la fila desde la cual se copian las entradas) en un flujo de bits con un código unario truncado u otros códigos. En otro ejemplo, el codificador de vídeo 20 puede codificar (y el decodificador de vídeo 30 puede decodificar) un valor de desplazamiento entre la línea actual y la línea predictiva de la correlación 240 utilizada como referencia para codificar la línea actual. Es decir, el codificador de vídeo 20 puede codificar una indicación de una línea (o columna) particular desde la cual se copia un valor de índice. En algunos ejemplos, el valor de desplazamiento puede ser un vector de desplazamiento. Es decir, sean $c[0]$, $c[1]$, ..., para denotar los índices de la línea actual de la correlación 240 y sean $u[0]$, $u[1]$, $u[2]$, ..., para denotar los índices de una línea predictiva de la correlación 240, tal como una línea vecina superior. En este ejemplo, dado un vector de desplazamiento d , el valor de índice para $c[i]$ puede predecirse a partir de $u[i+d]$. El valor de d puede codificarse utilizando códigos unarios, truncados, exponenciales-Golomb o de Golomb-Rice.

[0201] La figura 6 es un diagrama de flujo que ilustra un proceso ejemplar para codificar datos de vídeo usando una modalidad de codificación de paleta, congruente con las técnicas de esta divulgación. El procedimiento de la figura 6 se explica con respecto a un codificador de vídeo, tal como el codificador de vídeo 20 (Figuras 1 y 2) o el decodificador de vídeo 30 (Figuras 1 y 3). Sin embargo, debe entenderse que otros dispositivos de codificación de vídeo pueden configurarse para realizar un procedimiento similar. Además, ciertos pasos en el procedimiento pueden realizarse en un orden diferente o en paralelo. Del mismo modo, ciertos pasos pueden omitirse, y otros pasos pueden agregarse, en varios ejemplos.

[0202] Un codificador de vídeo, como el codificador de vídeo y/o el decodificador de vídeo 30 puede determinar inicialmente si una modalidad para codificar un bloque actual de datos de vídeo es una modalidad de codificación basada en paletas (280). Como se ha indicado anteriormente, uno o más elementos sintácticos pueden incluirse en un flujo de bits indicando una modalidad de codificación del bloque actual (por ejemplo, un elemento sintáctico indicador_Modalidad_PLT). En otros ejemplos, el codificador de vídeo puede tomar una determinación basándose en la llamada información lateral, como se ha señalado anteriormente.

[0203] En cualquier caso, si no se usa una modalidad de codificación basada en paletas para el bloque que se está codificando actualmente (la rama "no" de la etapa 280), el codificador de vídeo puede codificar el bloque de datos de vídeo usando una modalidad diferente a una modalidad de codificación basada en paletas (282). Por ejemplo, el codificador de vídeo puede codificar el bloque de datos de vídeo usando una modalidad no basada en paletas, tal como una intramodalidad, una intermodalidad u otra modalidad de codificación.

[0204] Si se utiliza una modalidad de codificación basada en paletas para el bloque que se está codificando actualmente (la rama "sí" de la etapa 280), el codificador de vídeo puede determinar una paleta para codificar el bloque actual (284). Como se ha descrito con respecto a las figuras 7 y 8 a continuación, en algunos ejemplos, el codificador de vídeo puede determinar la paleta para el bloque actual basándose en una paleta asociada a otros uno o más bloques de datos de vídeo previamente codificados.

[0205] El codificador de vídeo también puede determinar los valores de índices para el bloque que se está codificando actualmente (286). Por ejemplo, como se describe con mayor detalle con respecto a la figura 9 más adelante, el codificador de vídeo puede determinar una correlación que indica qué posiciones de píxel del bloque se codifican utilizando un valor de índice que relaciona una posición de un píxel con una entrada de la paleta determinada que tiene un valor de píxel asociado. En algunos casos, el codificador de vídeo puede determinar uno o más valores de índices relativos a otros valores de índices del bloque. Por ejemplo, el codificador de vídeo puede determinar una racha de valores de índices y/o un valor de índice basándose en un valor de índice ubicado en otra línea o columna del bloque.

[0206] El codificador de vídeo luego codifica los datos de vídeo del bloque usando la paleta determinada y los valores de índices (280). Por ejemplo, con respecto al codificador de vídeo 20, el codificador de vídeo 20 puede codificar datos que indican la paleta, así como los valores de índices, en un flujo de bits codificado. El codificador de vídeo 20 también puede codificar, para posiciones de píxel cualesquiera que no tengan un píxel correspondiente en la paleta determinada, los valores de píxel reales o sus versiones cuantizadas en tales posiciones. El decodificador de vídeo 30 puede analizar sintácticamente y decodificar la paleta, los valores de índices y los valores de píxeles de un flujo de bits codificado y utilizar los datos para reconstruir el bloque de datos de vídeo.

[0207] La figura 7 es un diagrama de flujo que ilustra un proceso ejemplar para determinar una paleta en la codificación basada en paletas, congruente con las técnicas de esta divulgación. El procedimiento de la figura 7 se explica con respecto a un codificador de vídeo, tal como el codificador de vídeo 20 (Figuras 1 y 2) o el decodificador de vídeo 30 (Figuras 1 y 3). Sin embargo, debe entenderse que otros dispositivos de codificación de vídeo pueden configurarse para realizar un procedimiento similar. Además, ciertos pasos en el procedimiento pueden realizarse en un orden diferente o en paralelo. Del mismo modo, ciertos pasos pueden omitirse, y otros pasos pueden agregarse, en varios ejemplos. En algunos ejemplos, las técnicas de la figura 7 se puede realizar durante la etapa 284 de la figura 6.

[0208] En el ejemplo de la figura 7, el codificador de vídeo puede determinar un tamaño de la paleta (300). En algunos ejemplos, el tamaño de la paleta puede ser fijo. En otros ejemplos, el tamaño de la paleta se puede ajustar dinámicamente durante la codificación (por ejemplo, agregando o eliminando entradas de la paleta). El codificador de vídeo puede codificar uno o más elementos sintácticos que indican el tamaño de la paleta.

[0209] El codificador de vídeo puede determinar si la paleta para codificar el bloque actual se predice a partir de otras una o más paletas previamente codificadas (302). Si se predice la paleta (la rama sí de la etapa 302), el codificador de vídeo puede determinar la paleta predictiva (304). Por ejemplo, el codificador de vídeo 20 puede codificar datos que indican que la paleta para el bloque actual se basa en una o más paletas codificadas previamente, así como datos que indican una ubicación de un bloque asociado a la paleta predictiva (o que identifican la propia paleta predictiva). Del mismo modo, el decodificador de vídeo 30 puede obtener dichos datos a partir de un flujo de bits codificado. Como se ha descrito anteriormente con respecto a la figura 4, la paleta predictiva puede asociarse a un bloque vecino del bloque que se está codificando actualmente.

[0210] El codificador de vídeo puede determinar una o más entradas de la paleta para codificar el bloque actual basándose en una o más entradas de la paleta predictiva determinada (306). En algunos ejemplos, el codificador de vídeo puede copiar una paleta completa de otro bloque para codificar el bloque actual. En otros ejemplos, el codificador de vídeo puede copiar selectivamente las entradas de otra paleta. La predicción de la paleta puede basarse, en algunos ejemplos, en función de una frecuencia con la que se utilizan las entradas de paleta en uno o más bloques codificados previamente.

[0211] El codificador de vídeo también puede determinar una o más entradas no predichas de la paleta (308). Por ejemplo, en algunos casos, solo una parte de una paleta puede predecirse a partir de otras paletas codificadas previamente. En otros casos, una paleta puede no estar codificada de manera predictiva en absoluto. En tales casos, el codificador de vídeo puede determinar entradas de paleta sin (o después de) realizar las técnicas de predicción de paleta descritas en el presente documento.

[0212] La figura 8 es un diagrama de flujo que ilustra un proceso ejemplar para determinar índices de un bloque de datos de vídeo, congruente con las técnicas de esta divulgación. El procedimiento de la figura 8 se explica con respecto a un codificador de vídeo, tal como el codificador de vídeo 20 (Figuras 1 y 2) o el decodificador de vídeo 30 (Figuras 1 y 3). Sin embargo, debe entenderse que otros dispositivos de codificación de vídeo pueden configurarse para realizar un procedimiento similar. Además, ciertos pasos en el procedimiento pueden realizarse en un orden diferente o en paralelo. Del mismo modo, ciertos pasos pueden omitirse, y otros pasos pueden agregarse, en varios ejemplos. En algunos ejemplos, las técnicas de la figura 8 se puede realizar durante la etapa 286 de la figura 6.

[0213] El codificador de vídeo puede determinar inicialmente una dirección de exploración, por ejemplo, para explorar una correlación de valores de índices en la codificación basada en paletas (320). En algunos ejemplos, como se ha indicado anteriormente con respecto al ejemplo de la figura 5, el codificador de vídeo puede estar preconfigurado para usar una dirección de exploración particular. En otros ejemplos, el codificador de vídeo puede codificar uno o más elementos sintácticos que indican la dirección de exploración y determinar la dirección de exploración según los elementos sintácticos. En otros ejemplos más, el codificador de vídeo puede determinar la dirección de exploración basándose en la llamada información lateral.

[0214] El codificador de vídeo también puede determinar una correlación de posiciones del bloque de datos de vídeo que se están codificando actualmente y que tienen valores de índices asociados de una paleta. Por ejemplo, uno o más valores de píxel del bloque que se está codificando actualmente pueden no estar representados en la paleta a partir del bloque. En tales ejemplos, el codificador de vídeo puede codificar los valores de píxeles directamente (en lugar de codificar un valor de índice que identifica el valor de píxel en la paleta). La correlación puede indicar qué posiciones de píxeles están asociadas a un valor de índice y qué posiciones de píxeles no están asociadas a un valor de índice.

[0215] El codificador de vídeo determina uno o más primeros valores de índice (324). Para un píxel dado en el bloque, el codificador de vídeo puede determinar un valor de índice que relaciona el valor de píxel del píxel dado con una entrada en la paleta para el bloque.

[0216] El codificador de vídeo también determina uno o más segundos valores de índices basándose en los primeros valores de índices (326). Por ejemplo, como se ha descrito anteriormente con respecto a la figura 5, el codificador de vídeo puede determinar uno o más valores de índices relativos a otros valores de índices ya codificados. En algunos ejemplos, el codificador de vídeo puede determinar segundos valores de índice basados en una racha de valores consecutivos de índices del mismo valor. En otros ejemplos, el codificador de vídeo puede determinar segundos valores de índices basándose en uno o más primeros valores de índices incluidos en una línea diferente a la línea de píxeles que se está codificando actualmente. En otros ejemplos, el codificador de vídeo puede aplicar una combinación de técnicas para determinar los valores del segundo índice.

[0217] Debe reconocerse que, dependiendo del ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, combinarse u omitirse por completo (por ejemplo, no todos los actos o sucesos descritos son necesarios para la puesta en práctica de las técnicas). Además, en determinados ejemplos, los actos o acontecimientos se pueden llevar a cabo simultáneamente, por ejemplo, a través de procesamiento de múltiples hilos, procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente. Además, si bien determinados aspectos de la presente divulgación se describen como que se llevan a cabo por un único módulo o unidad para propósitos de claridad, se debe entender que las técnicas de la presente divulgación se pueden llevar a cabo por una combinación de unidades o módulos asociados con un codificador de vídeo.

[0218] Ciertos aspectos de esta divulgación se han descrito con respecto a la norma de HEVC en desarrollo, con fines de ilustración. Sin embargo, las técnicas descritas en esta divulgación pueden ser útiles para otros procesos de codificación de vídeo, que incluyen otros procesos de codificación de vídeo, estándar o patentados, aún no desarrollados.

[0219] Las técnicas descritas anteriormente pueden realizarse mediante el codificador de vídeo 20 (FIGs. 1 y 2) y / o el decodificador de vídeo 30 (FIGs. 1 y 3), los cuales pueden mencionarse generalmente como un codificador de vídeo. Del mismo modo, la codificación de vídeo puede referirse a la codificación de vídeo o a la decodificación de vídeo, según corresponda.

[0220] Si bien se describen anteriormente combinaciones particulares de diversos aspectos de las técnicas, estas combinaciones se proporcionan meramente para ilustrar ejemplos de las técnicas descritas en esta divulgación. En consecuencia, las técnicas de esta divulgación no deberían limitarse a estas combinaciones a modo de ejemplo y pueden abarcar cualquier combinación concebible de los diversos aspectos de las técnicas descritas en esta divulgación.

[0221] 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 se pueden almacenar en, o transmitir a través de, un medio legible por ordenador como una o más instrucciones o código y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que correspondan a un medio tangible tal como medios de almacenamiento de datos o medios de comunicación que incluyan cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) medios de almacenamiento tangibles legibles por ordenador que sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser medios disponibles cualesquiera a los que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

[0222] A modo de ejemplo, y no de manera limitativa, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe adecuadamente la denominación de medios legibles por ordenador. Por ejemplo, si las instrucciones se transmiten desde una página web, un servidor u otra fuente remota usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, donde algunos discos normalmente emiten datos magnéticamente, mientras que otros discos emiten datos ópticamente con láseres. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

[0223] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables in situ (FPGA) u otros circuitos lógicos, integrados o discretos, equivalentes. En consecuencia, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar 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 se podrían implementar totalmente en uno o más circuitos o elementos lógicos.

[0224] Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un equipo manual inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto

de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec, o ser proporcionadas por un grupo de unidades de hardware interoperativas, incluyendo uno o más procesadores, como se ha descrito anteriormente, conjuntamente con software y/o firmware adecuados.

REIVINDICACIONES

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

- 5 determinar (304) una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxeles, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en entradas de paleta (202, 204, 206) asociadas a paletas de una pluralidad de bloques de datos de vídeo;
- 10 determinar (306), para una segunda paleta (192) asociada a un segundo bloque (188) de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta (208, 210, 212) de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y
- 15 decodificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la decodificación de los píxeles comprende:
- obtener (286), a partir de un flujo de bits codificado, respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índices identifican una entrada de la segunda paleta; y
- 20 determinar (288) valores para los uno o más píxeles apareando los respectivos valores de índices con al menos una de las entradas de la segunda paleta.

2. El procedimiento según la reivindicación 1, que comprende además:

- 25 cuando las una o más entradas de la segunda paleta incluyen todas las entradas de predictor de paleta, decodificar los datos que indican que las una o más entradas de la segunda paleta incluyen todas las entradas de predictor de paleta; y
- 30 cuando las una o más entradas de la segunda paleta incluyen menos que todas las entradas de predictor de paleta, decodificar, para cada entrada de la paleta predictora, datos que indican si la entrada respectiva está incluida en la segunda paleta.

3. El procedimiento según la reivindicación 2, que comprende además:

- 35 cuando las una o más entradas de la segunda paleta incluyen menos que todas las entradas de predictor de paleta, decodificar los datos que indican las entradas de paleta de la segunda paleta que no están incluidas en la paleta predictora.

4. El procedimiento según la reivindicación 1, que comprende además:

- 40 asignar valores de índices a las una o más entradas de la segunda paleta en función de la inclusión de las una o más entradas de la segunda paleta en la paleta predictora, incluida la asignación de un valor de índice relativamente más bajo, a las una o más entradas de la segunda paleta que estén incluidas en la paleta predictora, que otras entradas cualesquiera de la segunda paleta.

45 5. El procedimiento según la reivindicación 1, que comprende además:

- 50 decodificar un indicador de predicción para cada entrada de predictor de paleta de la paleta predictora, en donde el indicador de predicción indica si la respectiva entrada de predictor de paleta se incluye en la segunda paleta, y
- 55 en el que determinar las una o más entradas de paleta de la segunda paleta en función de las entradas de predictor de paleta de la paleta predictora comprende, para cada indicador de predicción de paleta que indica que la respectiva entrada de predictor de paleta está incluida en la segunda paleta, copiar la respectiva entrada de predictor de paleta en la segunda paleta.

6. El procedimiento según la reivindicación 1, que comprende además:

- 60 determinar una frecuencia con la que cada entrada de predictor de paleta de la paleta predictora se utiliza para codificar píxeles de la pluralidad de bloques; y

decodificar datos que identifican cada entrada de predictor de paleta de la paleta predictora en función de la frecuencia.

7. El procedimiento según la reivindicación 1, que comprende además:

decodificar un elemento sintáctico de predicción de paleta que indica que las una o más entradas de la segunda paleta se basan en las entradas de predictor de paleta.

8. El procedimiento de la reivindicación 1, que comprende además cuantizar las una o más entradas de la segunda paleta combinando entradas de la segunda paleta.

9. El procedimiento según la reivindicación 1, que comprende además:

agregar una entrada a la segunda paleta basándose en si una diferencia entre un valor de píxel de la entrada a agregar y los valores de píxeles de cada una de las una o más entradas de la segunda paleta es mayor que un valor delta.

10. El procedimiento de la reivindicación 1, que comprende además determinar un tamaño de la segunda paleta, en donde determinar el tamaño de la segunda paleta comprende al menos uno entre:

determinar que el tamaño de la segunda paleta es fijo;

determinar que el tamaño de la segunda paleta es variable y se basa en datos decodificados con el segundo bloque de datos de vídeo;

determinar el tamaño de la segunda paleta en función de una característica del segundo bloque de datos de vídeo; o

determinar el tamaño de la segunda paleta en función de los datos decodificados con el segundo bloque de datos de vídeo que indica una entrada final en la segunda paleta.

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

determinar (304) una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxeles, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en entradas de paleta (202, 204, 206) asociadas a paletas de una pluralidad de bloques de datos de vídeo;

determinar (306), para una segunda paleta (192) asociada a un segundo bloque (188) de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta (208, 210, 212) de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y

codificar los píxeles del segundo bloque de datos de vídeo usando la segunda paleta, en donde la codificación de los píxeles comprende:

determinar (286) respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índice identifican entradas respectivas de la segunda paleta; y

codificar (288) los valores de índices en un flujo de bits codificado.

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

medios para determinar (304) una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxeles, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en las entradas de paleta (202, 204, 206) asociadas a una o más paletas de una pluralidad de bloques de datos de vídeo;

medios para determinar (306), para una segunda paleta (192) asociada a un segundo bloque (188) de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta (208, 210, 212) de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y

medios para decodificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta,

en donde la decodificación de los píxeles comprende:

5 obtener (286), a partir de un flujo de bits codificado, respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índices identifican una entrada de la segunda paleta; y

 determinar (288) valores para los uno o más píxeles apareando los respectivos valores de índices con al menos una de las entradas de la segunda paleta.

10 **13.** El aparato de la reivindicación 12, que comprende medios para llevar a cabo un procedimiento de acuerdo a cualquiera de las reivindicaciones 2 a 10.

15 **14.** Un aparato para codificar datos de vídeo, comprendiendo el aparato:

 medios para determinar (304) una paleta predictora que tiene una pluralidad de entradas de predictor de paleta que indican respectivos valores de píxeles, en donde determinar la paleta predictora comprende determinar la paleta predictora basándose en las entradas de paleta (202, 204, 206) asociadas a una o más paletas de una pluralidad de bloques de datos de vídeo;

20 medios para determinar (306), para una segunda paleta asociada a un segundo bloque (188) de datos de vídeo que se codifica después de la pluralidad de bloques, una o más entradas de paleta (208, 210, 212) de la segunda paleta, basándose en las entradas de predictor de paleta de la paleta predictora; y

25 medios para codificar píxeles del segundo bloque de datos de vídeo usando la segunda paleta,

en donde la codificación de los píxeles comprende:

30 determinar (286) respectivos valores de índices para uno o más de los píxeles, en donde los respectivos valores de índice identifican entradas respectivas de la segunda paleta; y

 codificar (288) los valores de índices en un flujo de bits codificado.

35 **15.** Un medio no transitorio legible por ordenador que almacena instrucciones en el mismo que, cuando son ejecutadas, hacen que uno o más procesadores lleven a cabo un procedimiento de acuerdo a cualquiera de las reivindicaciones 1 a 11.

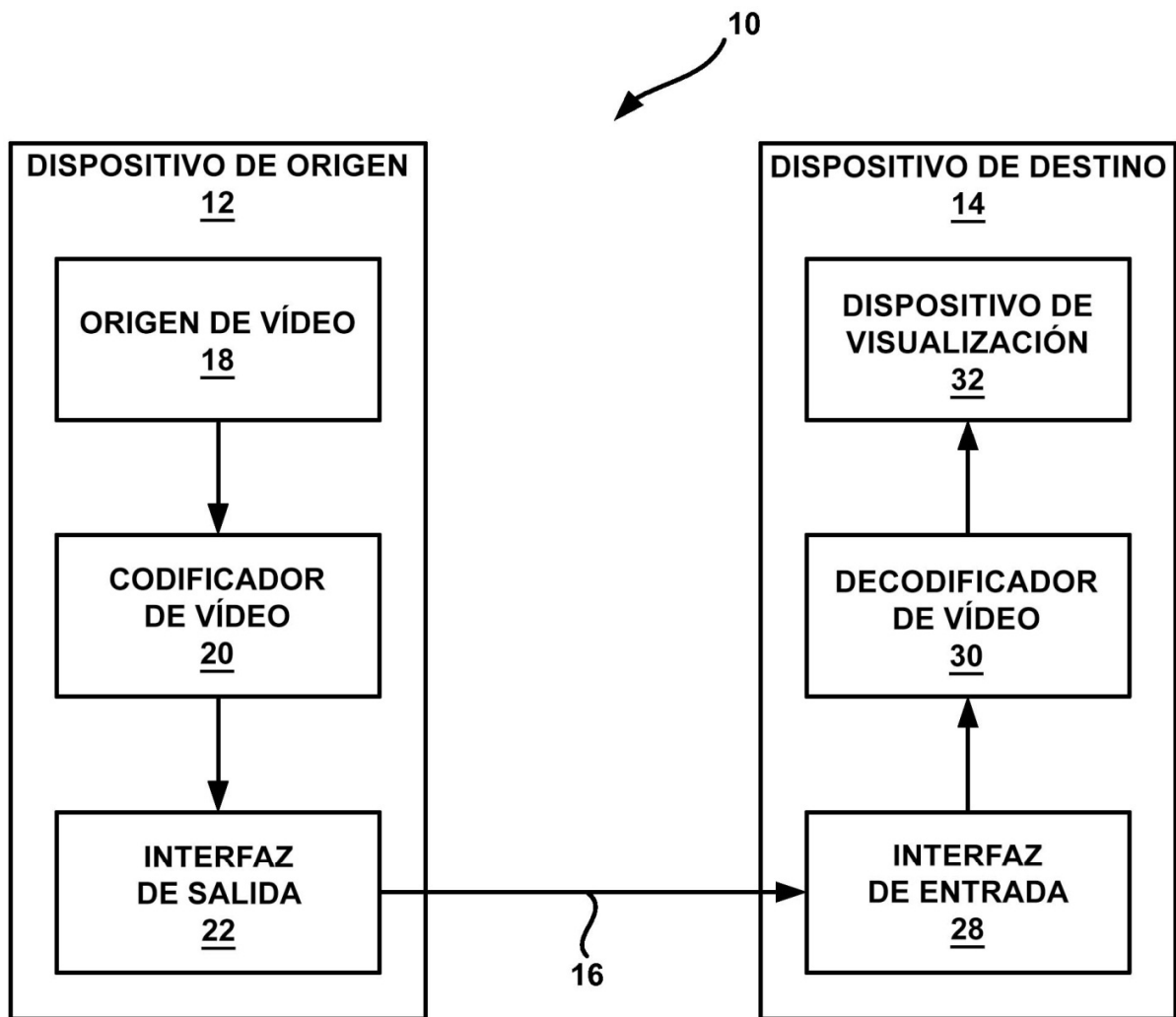


FIG. 1

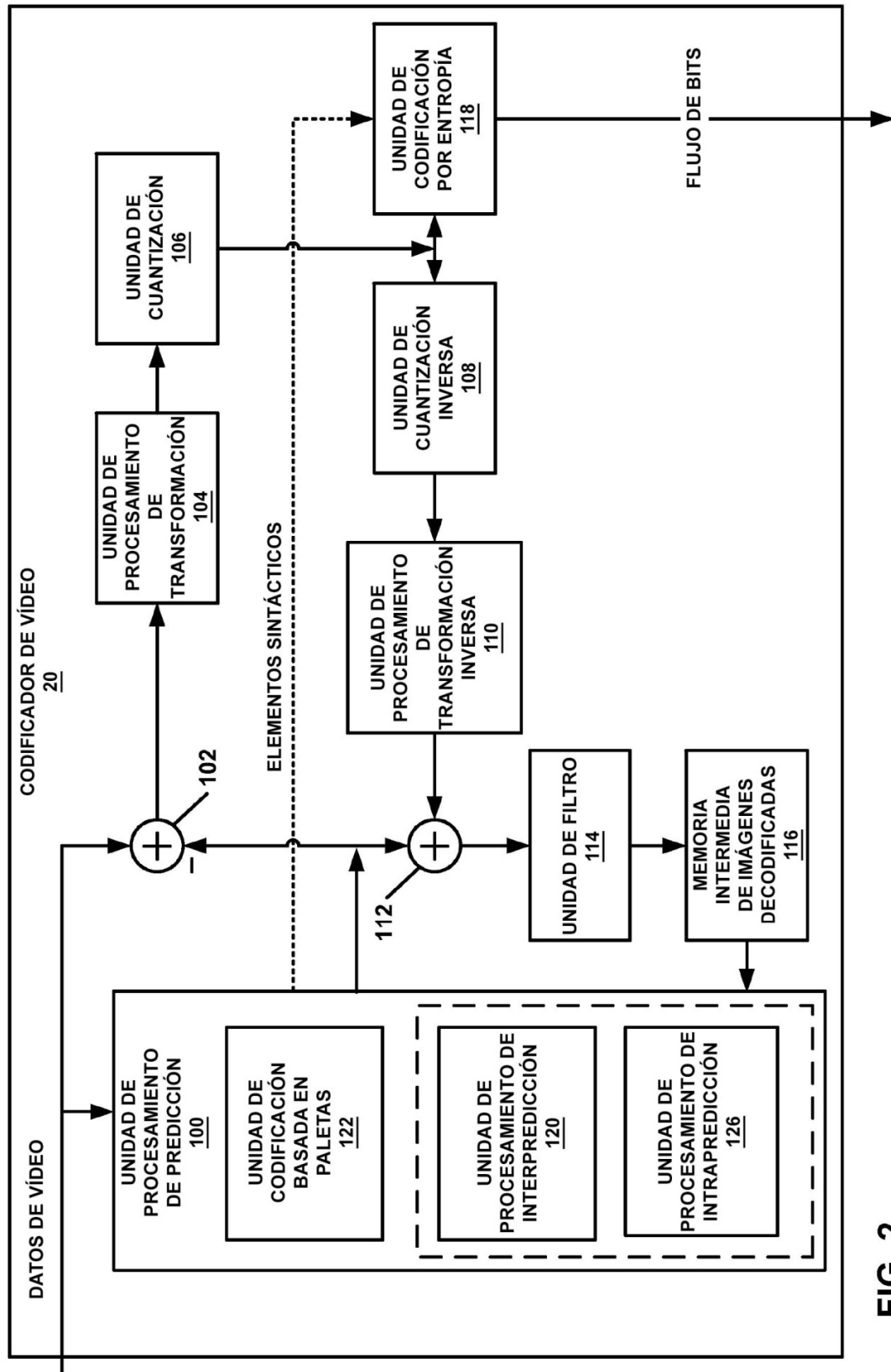


FIG. 2

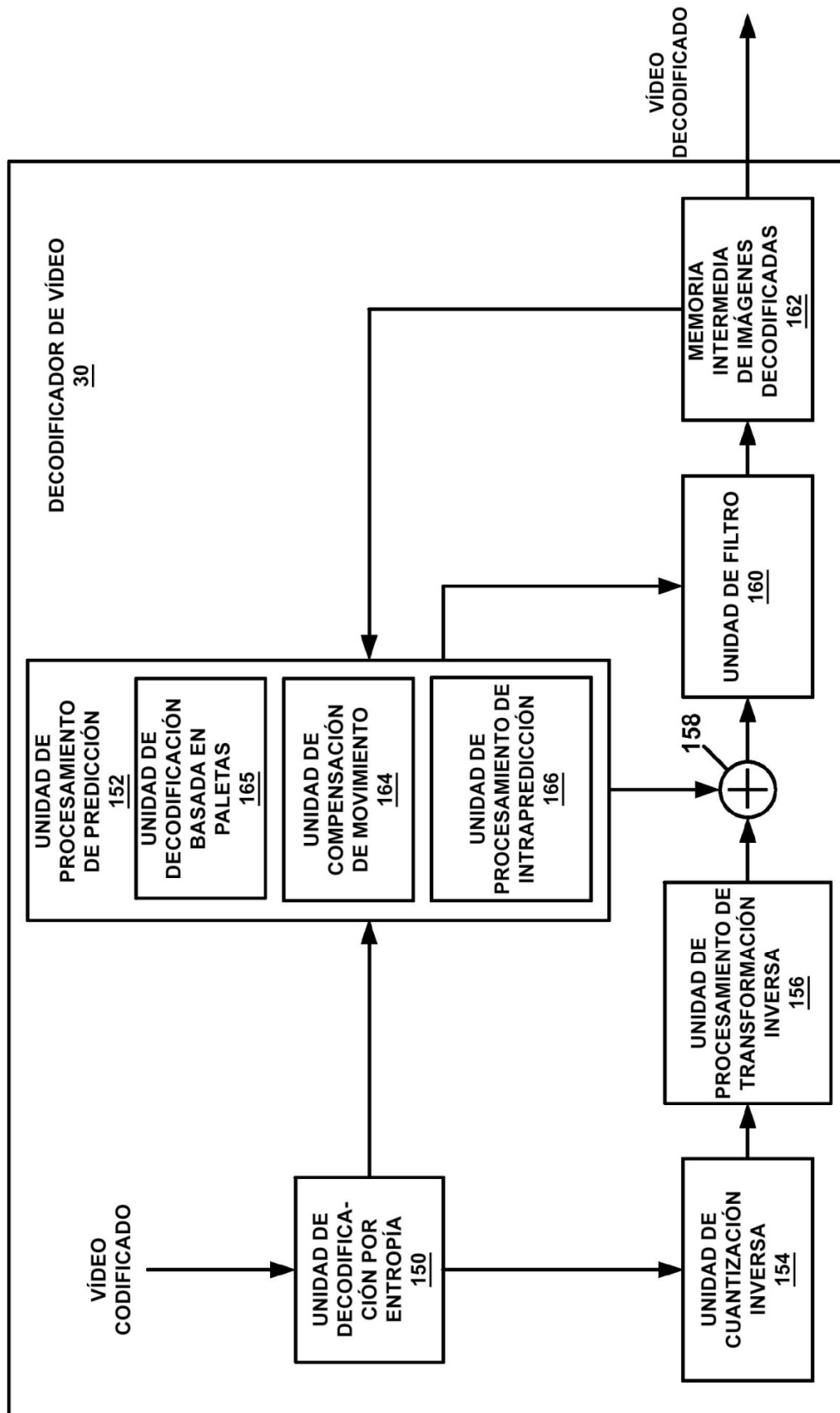


FIG. 3

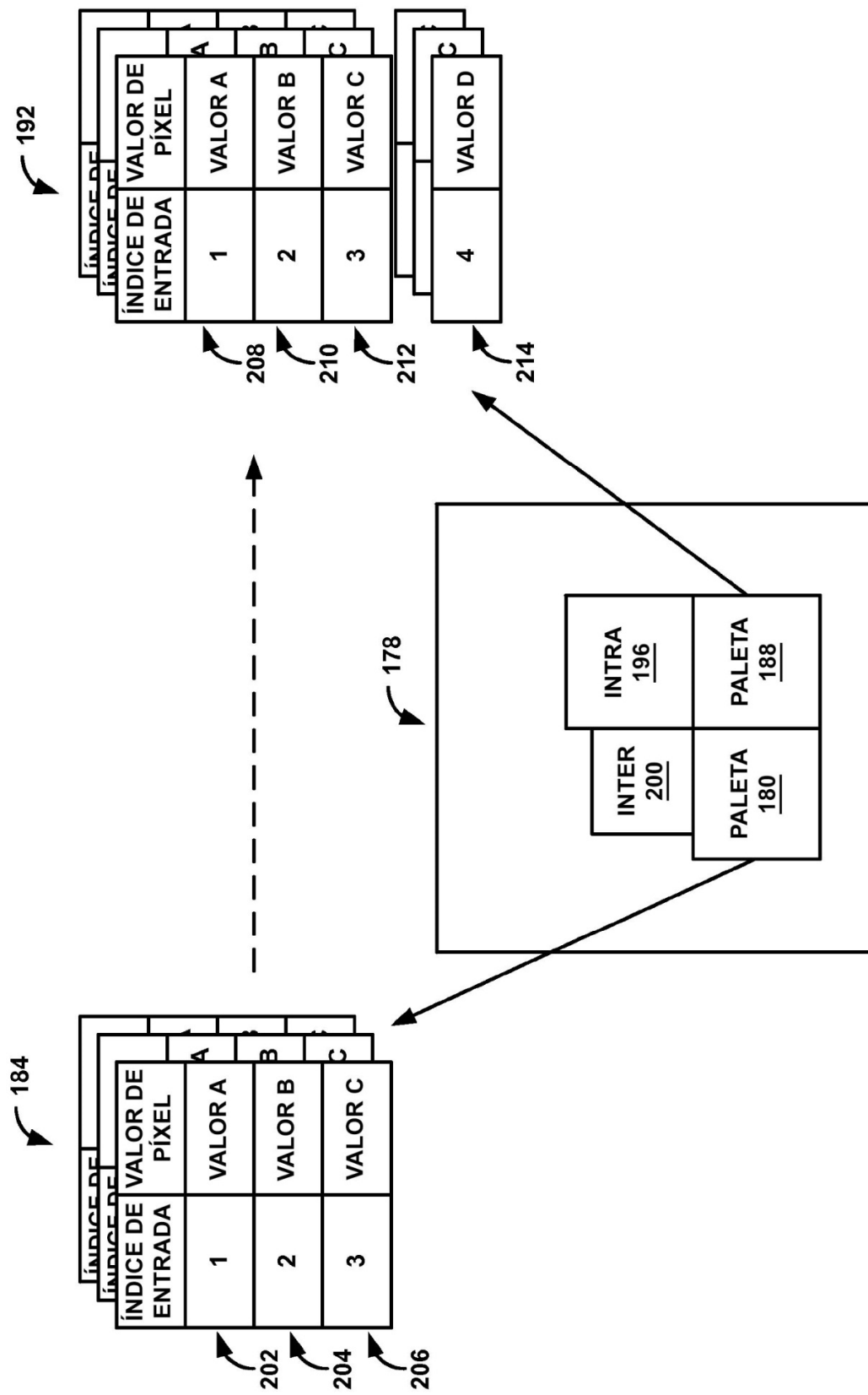


FIG. 4

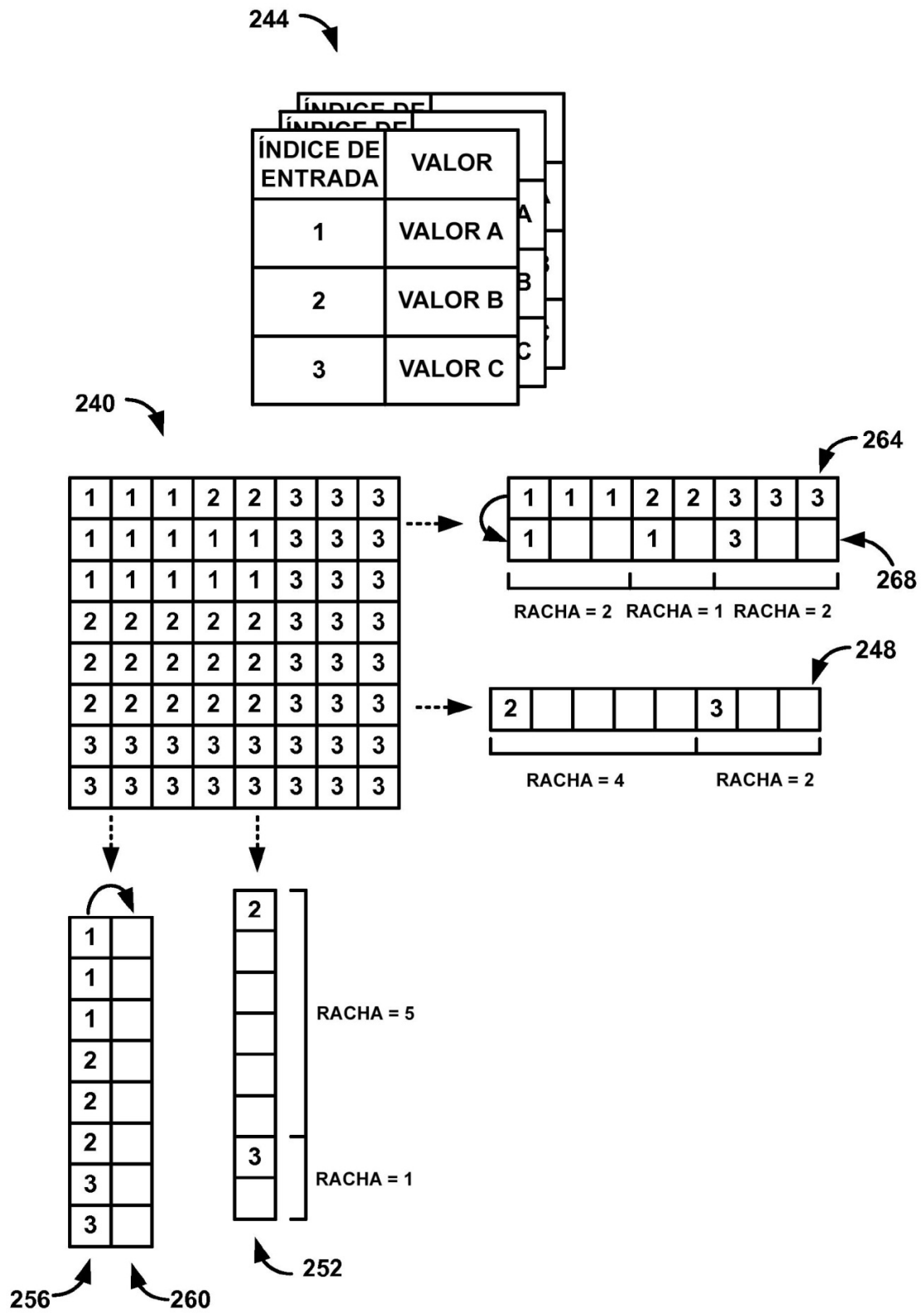


FIG. 5

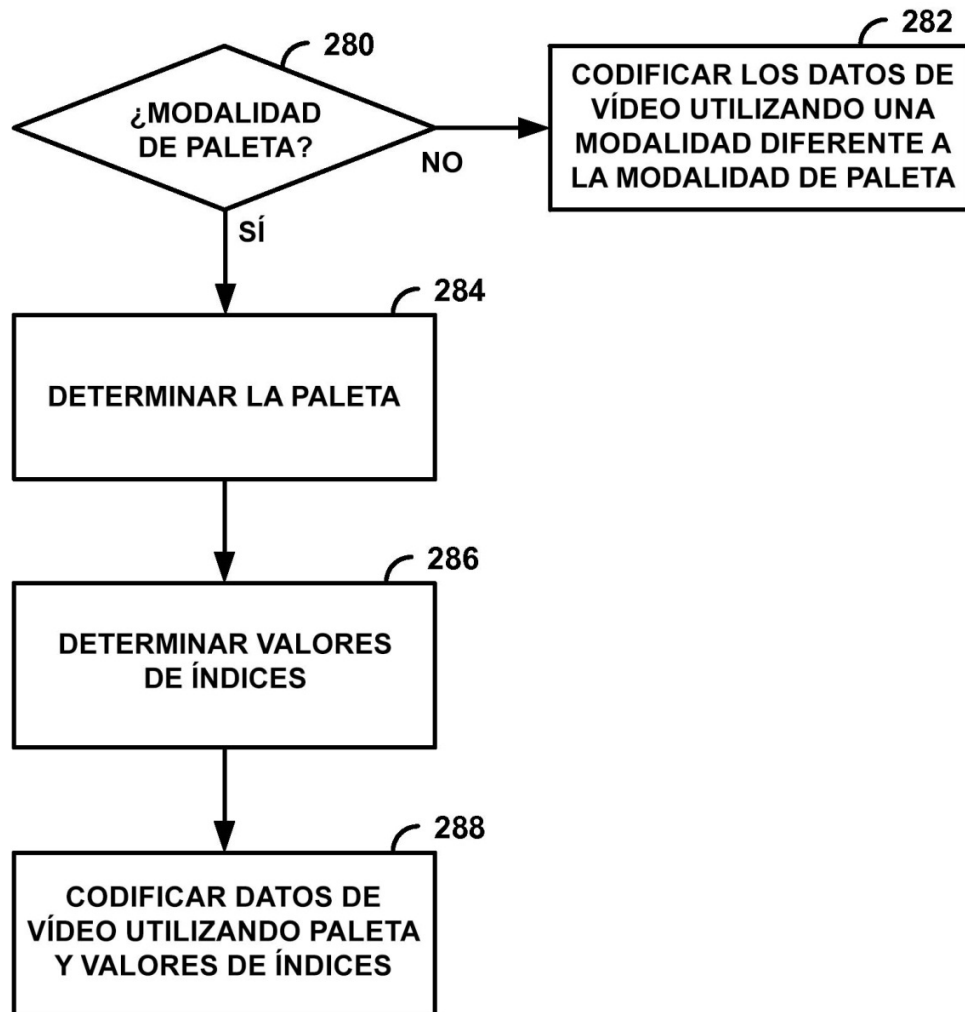


FIG. 6

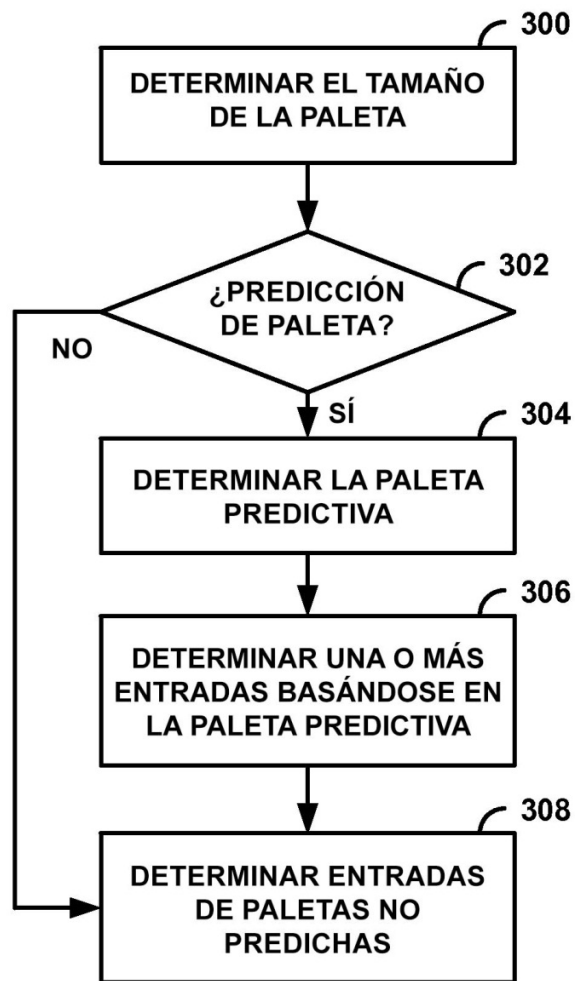


FIG. 7

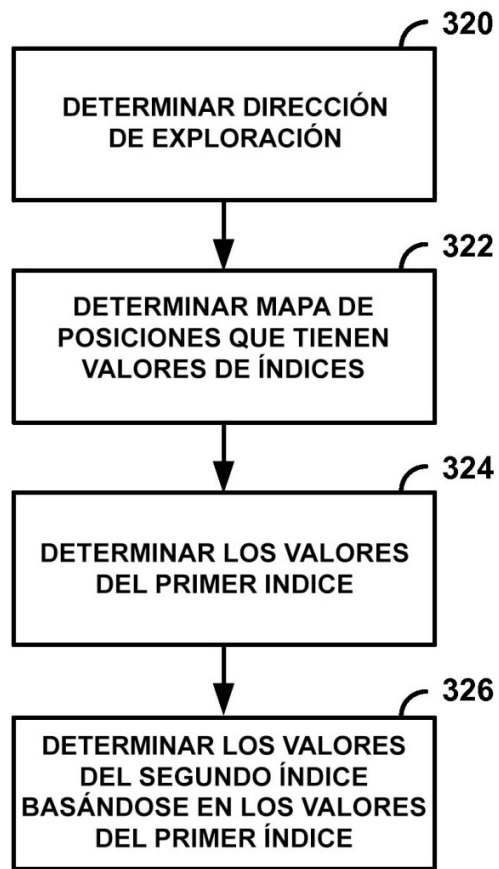


FIG. 8