

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 674 553**

51 Int. Cl.:

H04N 19/176 (2014.01)

H04N 19/105 (2014.01)

H04N 19/186 (2014.01)

H04N 19/157 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **22.05.2015 PCT/US2015/032262**

87 Fecha y número de publicación internacional: **26.11.2015 WO15179814**

96 Fecha de presentación y número de la solicitud europea: **22.05.2015 E 15729623 (7)**

97 Fecha y número de publicación de la concesión europea: **04.04.2018 EP 3146717**

54 Título: **Inicialización de la paleta predictora en codificación de vídeo basada en paletas**

30 Prioridad:

23.05.2014 US 201462002717 P

09.06.2014 US 201462009772 P

20.06.2014 US 201462015261 P

30.06.2014 US 201462019223 P

03.10.2014 US 201462059659 P

17.10.2014 US 201462065526 P

21.05.2015 US 201514719260

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
02.07.2018

73 Titular/es:

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

72 Inventor/es:

**SEREGIN, VADIM;
JOSHI, RAJAN LAXMAN;
KARCZEWICZ, MARTA;
PU, WEI y
SOLE ROJALS, JOEL**

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 674 553 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

Inicialización de la paleta predictora en codificación de vídeo basada en paletas

5 **[1]** Esta solicitud reivindica el beneficio de la Solicitud provisoria Estadounidense N° 62 / 002.717, presentada el 23 de mayo de 2014, la Solicitud Provisoria Estadounidense N° 62 / 009.772, presentada el 9 de junio de 2014, la Solicitud Provisoria Estadounidense N° 62 / 015.261, presentada el 20 de junio de 2014, la Solicitud Provisoria Estadounidense N° 62 / 019.223, presentada el 30 de junio de 2014, Solicitud Provisoria Estadounidense N° 62 / 059.659, presentada el 3 de octubre de 2014 y la Solicitud Provisoria Estadounidense N° 62 / 065.526, presentada el 17 de octubre de 2014.

CAMPO TÉCNICO

15 **[2]** Esta divulgación se refiere a la codificación y decodificación de vídeo.

ANTECEDENTES

20 **[3]** Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de difusión directa digital, sistemas de difusión 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, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión por flujo de vídeo y similares. Los dispositivos de vídeo digitales implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificación Avanzada de Vídeo (AVC), la norma de Codificación de Vídeo de Alta Eficacia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital más eficazmente, implementando dichas técnicas de compresión de vídeo.

30 **[4]** Las técnicas de compresión de vídeo realizan la predicción espacial (intra-imagen) y/o la predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca en las secuencias de vídeo. Para la codificación 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 intra-codificado (I) de una imagen se codifican mediante la predicción espacial con respecto a las muestras de referencia en bloques contiguos en la misma imagen. Los bloques de vídeo en un fragmento inter-codificado (P o B) de una imagen pueden usar la predicción espacial con respecto a las muestras de referencia en bloques contiguos en la misma imagen, o la predicción temporal con respecto a las muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

40 **[5]** La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo a 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 intra-codificado se codifica de acuerdo a una modalidad de intra-codificación y a los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio del píxel a un dominio de transformación, dando como resultado coeficientes residuales, los cuales pueden cuantizarse posteriormente. Los coeficientes cuantizados, inicialmente dispuestos en una formación bidimensional, pueden recorrerse con el fin de producir un vector unidimensional de coeficientes, y puede aplicarse la codificación por entropía para lograr aún más compresión.

SUMARIO

55 **[6]** Las técnicas de esta divulgación se refieren a la codificación de vídeo basada en paletas. El alcance de la protección está definido por las reivindicaciones adjuntas a la presente.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

60 **[7]** La FIG. 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 FIG. 2 es un diagrama de bloques que ilustra un codificador de vídeo ejemplar que puede implementar las técnicas descritas en esta divulgación.

65

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

5 La FIG. 4 es un diagrama conceptual que ilustra un ejemplo de determinación de entradas de paleta para la codificación de vídeo basada en paletas, congruente con las técnicas de esta divulgación.

La FIG. 5 es un diagrama conceptual que ilustra teselas ejemplares de acuerdo a la norma de codificación de vídeo de alta eficacia (HEVC).

10 La FIG. 6 es un diagrama conceptual que ilustra los frentes de onda para el procesamiento paralelo de frentes de onda (WPP).

15 La FIG. 7 es un diagrama conceptual que ilustra un ejemplo de reinicio de paleta predictora en presencia de múltiples teselas y frentes de onda, congruente con las técnicas de esta divulgación.

La FIG. 8 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.

20 La FIG. 9 es un diagrama de flujo que ilustra un proceso ejemplar para la determinación de una paleta predictora, congruente con las técnicas de esta divulgación.

La FIG. 10 es un diagrama de flujo que ilustra un proceso ejemplar para codificar un valor de racha de una racha de índices de paleta, congruente con las técnicas de esta divulgación.

25 La FIG. 11 es un diagrama de flujo que ilustra un proceso ejemplar para codificar una racha de índices codificados con más de una modalidad de codificación de paleta, congruente con las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

30 **[8]** Los aspectos de esta divulgación se orientan a técnicas para la codificación de vídeo y la compresión de datos 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. En función de estas hipótesis, se han desarrollado varias herramientas, tales como la transformación basada
35 en bloques, el filtrado y otras herramientas de codificación, y dichas herramientas han demostrado un buen rendimiento para los vídeos de contenido natural.

[9] Sin embargo, en aplicaciones como el escritorio remoto, el trabajo colaborativo y la pantalla inalámbrica, el contenido de pantalla generado por ordenador puede ser el contenido dominante a comprimir. Este tipo de contenido
40 tiende a tener un tono discreto, incluir líneas nítidas y presentar límites de objetos de alto contraste. La hipótesis de tono continuo y fluidez puede no valer ya y, por lo tanto, las técnicas tradicionales de codificación de vídeo pueden ser ineficaces para comprimir el contenido.

[10] Esta divulgación describe la codificación basada en paletas, que puede ser particularmente adecuada para la codificación de contenido generado por pantalla u otro contenido en el que una o más herramientas de
45 codificación tradicionales son ineficaces. 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 inter-predictiva o intra-predictiva. 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
50 inter-predictiva e intra-predictiva, así como una codificación basada en paletas.

[11] En algunos ejemplos, las técnicas de codificación basadas en paletas pueden configurarse para su uso con una o más normas de codificación de vídeo. Además, la codificación de vídeo de alta eficacia (HEVC) es una nueva norma de codificación de vídeo que está siendo desarrollada por el Equipo de Colaboración Conjunta en
55 Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de ISO/IEC. Un 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)"], JVCV-L1003_v13, 12a conferencia del JCT-VC de ITU-T SG16 WP 3 e ISO / IEC
60 JCT 1 / SC 29 / WG 11, 14 al 23 de enero de 2013 ("Borrador 10 de HEVC").

[12] Con respecto al entorno de la HEVC, como ejemplo, las técnicas de codificación basadas en paletas se pueden configurar para usarse como una modalidad de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basadas en paletas pueden configurarse para usarse como una modalidad de PU en el entorno de la HEVC. En consecuencia, todos los procesos divulgados a continuación, descritos en el contexto de una modalidad 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 basadas en paletas, descritas en el presente documento, ya que tales 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 paletas pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[13] En la codificación basada en paletas se puede suponer 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 codificar 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.

[14] En el ejemplo anterior, un codificador de vídeo 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. Un decodificador de vídeo 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 puede relacionar los valores de índice de los píxeles con las entradas de la paleta para reconstruir los valores de píxel del bloque. Los píxeles (y / o valores de índice relacionados que indican un valor de píxel) generalmente se pueden denominar muestras.

[15] Se supone que las muestras en el bloque se procesan (por ejemplo, se escanean) usando el orden de escaneo de cuadrado horizontal. Por ejemplo, el codificador de vídeo puede convertir un bloque bidimensional de índices en una formación unidimensional escaneando los índices usando un orden de escaneo de cuadrado horizontal. Del mismo modo, el decodificador de vídeo puede reconstruir un bloque de índices utilizando el orden de escaneo de cuadrado horizontal. Por consiguiente, esta divulgación puede referirse a una muestra previa como una muestra que precede a la muestra que se está codificando actualmente en el bloque en el orden de escaneo. Se debería tener en cuenta que los escaneos que no sean un escaneo de cuadrado horizontal, tales como el orden de escaneo de cuadrado vertical, también pueden ser aplicables. El ejemplo anterior pretende proporcionar una descripción general de la codificación basada en paletas.

[16] Una paleta incluye típicamente entradas numeradas por un índice y que representan al menos valores o intensidades de un componente de color (por ejemplo, al menos en un componente de RGB, YUV o similares). Tanto un codificador de vídeo como un decodificador de vídeo determinan el número de entradas de paleta, los valores de los componentes de color para cada entrada de paleta y el ordenamiento exacto de las entradas de paleta para el bloque actual. En esta divulgación, se supone que cada entrada de paleta especifica los valores para todos los componentes de color de una muestra. Sin embargo, los conceptos de esta divulgación son aplicables al uso de una paleta independiente para cada componente de color.

[17] En algunos ejemplos, una paleta se puede componer utilizando información de bloques previamente codificados o paletas codificadas previamente. Es decir, una paleta puede contener entradas de paleta predichas a partir de la(s) paleta(s) utilizada(s) para codificar el(los) bloque(s) previo(s). Por ejemplo, como se describe en el documento de presentación estándar de Wei Pu et al., "AHG10: Software sugerido para la codificación de paletas basada en RExt6.0", JCTVC-Q0094, Valencia, ES, del 27 de marzo al 4 de abril de 2014 (en adelante, JCTVC-Q0094), una paleta puede incluir entradas que se copian desde una paleta predictora. Una paleta predictora puede incluir entradas de paleta de bloques codificados previamente usando la modalidad de paleta u otras muestras reconstruidas. La paleta predictora puede usar la totalidad o parte de una paleta previamente codificada, o puede estar compuesta de entradas de varias paletas codificadas previamente.

[18] En algunos ejemplos, para cada entrada en la paleta predictora, se puede codificar un indicador binario para indicar si la entrada asociada al indicador se copia a la paleta actual (por ejemplo, se indica mediante indicador = 1). La cadena de indicadores binarios se puede denominar vector de predicción de paleta binaria. La paleta para codificar un bloque actual también puede incluir un cierto número de nuevas entradas de paleta, que pueden codificarse explícitamente (por ejemplo, por separado del vector de predicción de paleta). También se puede codificar una indicación del número de nuevas entradas. Una suma de las entradas predichas y las nuevas entradas puede indicar el tamaño total de la paleta en el bloque.

[19] Como se propone en el documento JCTVC-Q0094, cada muestra en un bloque codificado con una modalidad de codificación basada en paletas se puede codificar usando una de las tres modalidades de paleta, como se expone a continuación:

- Modalidad de escape: en esta modalidad, el valor de muestra no se incluye en una paleta como una entrada de paleta y el valor de muestra cuantizado se señala explícitamente para todos los componentes de color. Es similar a la señalización de las nuevas entradas de paleta, aunque para las nuevas entradas de paleta, los valores de los componentes de color no están cuantizados.

- Modalidad CopiarDesdeArriba (también denominada modalidad CopiarArriba): en esta modalidad, el índice de entrada de paleta para la muestra actual se copia desde la muestra ubicada directamente arriba en un bloque.

- 5 • Modalidad de valor (también denominada modalidad de índice): en esta modalidad, el valor del índice de entrada de paleta se señala explícitamente.

10 **[20]** Como se describe en el presente documento, un índice de entrada de paleta se puede mencionar como un índice de paleta o simplemente un índice. Estos términos pueden usarse indistintamente para describir las técnicas de esta divulgación. Además, como se describe en mayor detalle a continuación, un índice de paleta puede tener uno o más valores de color o intensidad asociados. Por ejemplo, un índice de paleta puede tener un único valor asociado de color o intensidad, asociado a un único componente de color o intensidad de un píxel (por ejemplo, un componente rojo de datos RGB, un componente Y de datos YUV o similares). En otro ejemplo, un índice de paleta puede tener múltiples valores de color o intensidad asociados. En algunos casos, la codificación basada en paletas se puede aplicar para codificar vídeo monocromo. Por consiguiente, el "valor de color" puede referirse en general a cualquier componente de color, o no de color, utilizado para generar un valor de píxel.

20 **[21]** Para las modalidades de CopiarDesdeArriba y de Valor, también se puede señalar un valor de racha (que también puede denominarse simplemente racha). Un valor de racha puede indicar un cierto número de muestras consecutivas (por ejemplo, una racha de muestras) en una orden de escaneo particular en un bloque codificado con paletas que se codifican juntas. En algunos casos, la racha de muestras también se puede denominar una racha de índices de paleta, porque cada muestra de la racha tiene un índice asociado a una paleta.

25 **[22]** Un valor de racha puede indicar una racha de índices de paleta que están codificados usando la misma modalidad de codificación de paleta. Por ejemplo, con respecto a la modalidad de Valor, un codificador de vídeo (codificador de vídeo o decodificador de vídeo) puede codificar un índice de paleta (también denominado valor de índice de paleta o simplemente valor de índice) y un valor de racha que indica un cierto número de muestras consecutivas en un orden de escaneo que tienen el mismo índice de paleta y que están siendo codificadas con el índice de paleta. Con respecto a la modalidad CopiarDesdeArriba, el codificador de vídeo puede codificar una indicación de que un índice para el valor de muestra actual se copia basándose en un índice de una muestra vecina por arriba (por ejemplo, una muestra que está situada encima de la muestra que está siendo codificada actualmente en un bloque) y un valor de racha que indica un número de muestras consecutivas en un orden de escaneo que también copian un índice de paleta de una muestra vecina por arriba y que se están codificando con el índice de paleta. En consecuencia, en los ejemplos anteriores, una racha de índices de paleta se refiere a una racha de índices de paleta que tienen el mismo valor o una racha de índices de paleta que se copian de los índices de paleta vecinos por arriba.

40 **[23]** Por lo tanto, la racha puede especificar, para una modalidad determinada, el número de muestras subsiguientes que pertenecen a la misma modalidad. En algunos casos, la señalización de un índice y un valor de racha puede ser similar a la codificación de longitud de racha. En un ejemplo con fines de ilustración, una cadena de índices consecutivos de un bloque puede ser 0, 2, 2, 2, 5 (por ejemplo, donde cada índice corresponde a una muestra en el bloque). En este ejemplo, un codificador de vídeo puede codificar la segunda muestra (por ejemplo, el primer valor de índice igual a dos) usando la modalidad de Valor. Después de codificar un índice que es igual a 2, el codificador de vídeo puede codificar una racha de tres, lo que indica que las tres muestras subsiguientes también tienen el mismo valor de índice igual a dos. De manera similar, codificar una racha de cuatro después de codificar un índice usando la modalidad CopiarDesdeArriba puede indicar que se copian un total de cinco índices de los índices correspondientes en la fila encima de la posición de muestra que se está codificando actualmente.

50 **[24]** Como se describe en mayor detalle a continuación, un codificador de vídeo (por ejemplo, un codificador de vídeo y un decodificador de vídeo) puede codificar o decodificar datos que indican si una muestra se codifica como una muestra de escape muestra por muestra. Las muestras de escape (también denominadas píxeles de escape) pueden ser muestras (o píxeles) de un bloque que no tienen un color correspondiente representado en una paleta para codificar el bloque. En consecuencia, las muestras de escape no pueden reconstruirse utilizando una entrada de color (o valor de píxel) de una paleta. En cambio, los valores de color para las muestras de escape se señalizan en un flujo de bits por separado de los valores de color de la paleta. En general, codificar una muestra usando la "modalidad de Escape" puede referirse en general a la codificación de una muestra de un bloque que no tiene un color correspondiente representado en una paleta para codificar el bloque. Como se ha indicado anteriormente, tales muestras pueden denominarse muestras de escape o píxeles de escape.

60 **[25]** En algunos ejemplos, el codificador de vídeo puede codificar un indicador para cada muestra que indica si la muestra está codificada como una muestra de escape (esta técnica puede denominarse señalización de escape explícita, como se describe con mayor detalle a continuación con respecto al ejemplo de la FIG. 1). En otro ejemplo, el codificador de vídeo puede codificar un valor de índice adicional para una paleta, para indicar que una muestra particular está codificada como una muestra de escape (esta técnica puede denominarse señalización de escape implícita, como se describe en mayor detalle con respecto al ejemplo de la FIG. 1).

[26] Las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de predicción de entradas de paleta, codificación de rachas de índices de paleta y varias otras técnicas de codificación de paleta. Como se describe con mayor detalle a continuación, las técnicas de esta divulgación pueden, en algunos casos, mejorar la eficacia y mejorar la tasa de bits cuando se codifican datos de vídeo usando una modalidad de paleta.

[27] Por ejemplo, ciertos aspectos de esta divulgación están dirigidos a técnicas para predecir entradas de paleta para un bloque de datos de vídeo. En algunos casos, una paleta predictora puede reiniciarse para un bloque en el borde izquierdo de una imagen. Es decir, cuando se genera una paleta para el bloque en el borde izquierdo de la imagen, la paleta predictora puede reiniciarse en cero (por ejemplo, la paleta predictora no tiene entradas y la paleta actual no se predice utilizando una paleta predictora). La predictora puede reiniciarse, de esta manera, porque una paleta predictora puede incluir entradas de paleta que pertenecen mayormente a paletas de bloques ubicados en el lado derecho de la imagen después de decodificar una línea (p. ej., suponiendo un escaneo de cuadrulado de izquierda a derecha). En consecuencia, al codificar el primer bloque de la fila siguiente (el bloque en el borde más a la izquierda de la imagen), la paleta predictora puede incluir colores de bloques que están ubicados relativamente lejos del bloque que se está codificando actualmente. Por lo tanto, la paleta predictora puede no ser muy efectiva para predecir una paleta actual para el bloque actual (por ejemplo, los colores de los píxeles en el lado izquierdo de una imagen pueden diferir de los colores de los píxeles en el lado derecho de la imagen).

[28] Sin embargo, en algunos casos, reiniciar una paleta predictora puede conducir a pérdidas de codificación. Por ejemplo, sin una paleta predictora, un codificador de vídeo (un codificador de vídeo 20 o decodificador de vídeo) puede codificar datos que indican todas las entradas de una paleta (por ejemplo, todos los índices de paleta y valores de color relacionados) en un flujo de bits. Esta puede ser una cantidad relativamente grande de datos relativos a los datos asociados a las entradas de paleta predichas. En consecuencia, reiniciar una paleta predictora puede afectar negativamente a la tasa de bits de los datos de vídeo codificados con codificación basada en paletas.

[29] De acuerdo a aspectos de esta divulgación, una paleta predictora para construir una paleta para un bloque de datos de vídeo en una primera línea puede reinicializarse basándose en uno o más bloques de otra línea. Por ejemplo, un codificador de vídeo puede determinar una primera paleta para un primer bloque de datos de vídeo que está ubicado en una primera fila de bloques. El codificador de vídeo también puede generar una paleta predictora cuando codifica otros uno o más bloques en la primera fila. Al codificar un bloque en una segunda fila, el codificador de vídeo puede reinicializar la paleta predictora para determinar una paleta del bloque en la segunda fila basándose en las entradas de la primera paleta. De acuerdo a algunos aspectos de esta divulgación, la fila de bloques puede ser una fila de bloques de un cierto tamaño (por ejemplo, una fila de unidades de árbol de codificación (CTU), como se describe a continuación). La longitud de la fila puede representar un ancho de imagen en las unidades de bloque seleccionadas, y el número de filas puede representar la altura de la imagen en las unidades de bloque seleccionadas.

[30] En algunos ejemplos, la paleta predictora reinicializada incluye entradas de una paleta de uno o más bloques que están situados relativamente cerca del bloque que se está codificando actualmente. Por consiguiente, la paleta predictora puede incluir entradas que tienen una mayor probabilidad de ser incluidas en una paleta del bloque que se está codificando actualmente (por ejemplo, en relación con una paleta predictora basada en bloques ubicados lejos del bloque actual o una paleta predictora que se ha reiniciado). De esta manera, las técnicas de esta divulgación pueden aumentar la eficacia de la codificación, porque el codificador de vídeo puede determinar la paleta para el bloque actual utilizando la paleta predictora en lugar de codificar las entradas de la paleta en el flujo de bits.

[31] Otros aspectos de esta divulgación están dirigidos a la codificación (es decir, codificación o decodificación) de un valor de racha que indica una longitud de racha de una racha de índices de paleta. Por ejemplo, como se ha indicado anteriormente, una racha puede especificar, para un índice que actualmente se está codificando con una modalidad determinada, el número de muestras posteriores que se codifican con el índice actual utilizando la misma modalidad.

[32] En algunos casos, los datos que indican el valor de racha pueden codificarse utilizando una técnica de codificación adaptativa al contexto, tal como la codificación aritmética binaria adaptativa al contexto (CABAC), la codificación de longitud variable adaptativa al contexto (CAVLC) u otra técnica de codificación adaptativa al contexto. Por ejemplo, un codificador de vídeo (un codificador de vídeo o un decodificador de vídeo) puede seleccionar un modelo de probabilidad o "modelo de contexto" que opera sobre el contexto para codificar símbolos asociados a un bloque de datos de vídeo. Es decir, el modelo de contexto (Ctx) puede ser un índice o desplazamiento que se aplica para seleccionar uno entre una pluralidad de contextos diferentes, cada uno de los cuales puede corresponder a un modelo de probabilidad particular.

[33] En algunos casos, se puede usar un solo contexto para codificar datos que indiquen un valor de racha. Por ejemplo, un valor de racha binarizado puede incluir un primer contenedor que indica si el valor de racha es mayor que cero, un segundo contenedor que indica si el valor de racha es mayor que uno, un tercer contenedor que indica si el valor de racha es mayor que dos y otros contenedores cualesquiera, necesarios para representar el valor de racha. En este ejemplo, el mismo contexto se puede usar para codificar por contexto los primeros tres contenedores

del valor de racha binarizado. Sin embargo, usar el mismo modelo de probabilidad para codificar contenedores múltiples puede crear un retraso entre ciclos de codificación sucesivos. Además, la correlación de los contenedores de un valor de racha puede no ser suficiente para garantizar el tiempo y los recursos informáticos asociados a la actualización del modelo de probabilidad.

[34] De acuerdo a aspectos de esta divulgación, el contexto puede seleccionarse basándose en el valor de índice del valor de racha que se está codificando. Por ejemplo, un codificador de vídeo puede determinar un valor de racha que indica una longitud de racha de una racha de un índice de paleta de un bloque de datos de vídeo. El codificador de vídeo también puede determinar un contexto para los datos de codificación adaptativa al contexto que representan el valor de racha basándose en el valor del índice de la paleta. Las técnicas pueden mejorar la eficacia de codificación.

[35] En algunos ejemplos, el índice de paleta utilizado para obtener contextos para la codificación de rachas puede ser un índice de paleta que se utiliza para recuperar un valor de color de la paleta. En otros ejemplos, el índice de paleta utilizado para obtener contextos para la codificación de rachas puede ser un índice de paleta analizada, es decir, el índice de paleta señalado en un flujo de bits (que puede ser diferente al índice de paleta que se utiliza para acceder a los colores de la paleta, como se describe en mayor detalle a continuación).

[36] En algunos ejemplos, se pueden usar tres contextos para codificar un valor de racha. En dichos ejemplos, de acuerdo a aspectos de esta divulgación, un codificador de vídeo puede seleccionar un primer contexto para codificar el valor de racha basándose en el índice que es mayor que cero. El codificador de vídeo puede seleccionar un primer contexto para codificar el valor de racha en función de que el índice sea mayor que cero. El codificador de vídeo puede seleccionar un segundo contexto para codificar el valor de racha en función de que el índice sea mayor que uno. El codificador de vídeo puede seleccionar un tercer contexto para codificar el valor de racha en función de que el índice sea mayor que dos. Con respecto al ejemplo de contenedor codificado de tres contextos, descrito anteriormente, el codificador de vídeo puede seleccionar los tres contextos para codificar cualquier combinación de los tres contenedores. Si bien el ejemplo anterior se describe con respecto a la definición de tres contextos asociados a tres características del valor de índice, debería entenderse que las técnicas descritas en este documento pueden extenderse para definir otros números de contextos basados en otras características del valor de índice.

[37] Otros aspectos de esta divulgación están dirigidos a codificar rachas de índices de paleta que están codificados usando más de una modalidad de paleta. En general, los índices que se codifican utilizando diferentes modalidades de paleta no se pueden codificar en la misma racha. En un ejemplo con fines ilustrativos, una racha codificada utilizando la modalidad CopiarDesdeArriba puede no incluir ningún índice que esté codificado como muestra de escape. En este ejemplo, un píxel codificado como una muestra de escape puede terminar una racha, lo que puede afectar adversamente la eficacia de la codificación debido a rachas relativamente más cortas para un bloque dado.

[38] De acuerdo a aspectos de esta divulgación, una racha puede incluir muestras que están codificadas usando más de una modalidad de paleta. Por ejemplo, un codificador de vídeo puede determinar índices de paleta de una primera fila de un bloque de datos de vídeo, donde los índices de paleta incluyen uno o más índices que están asociados a un valor de color en la paleta y un elemento sintáctico que no está asociado a un valor de color en la paleta (que puede denominarse índice de paleta a pesar de que el elemento sintáctico no corresponde necesariamente a un valor de índice). El codificador de vídeo también puede codificar una racha de índices de paleta de una segunda fila del bloque de datos de vídeo, relativos a los índices de paleta de la primera fila, donde la racha incluye los uno o más índices que están asociados a un valor de color en la paleta y el elemento sintáctico que no está asociado a un valor de color en la paleta.

[39] En algunos ejemplos, la racha de los índices de paleta puede incluir píxeles que están codificados con la modalidad CopiarDesdeArriba y también como muestras de escape, por ejemplo, usando la modalidad de Escape. Por ejemplo, los índices que están asociados a un valor de color en la paleta se pueden codificar con la modalidad CopiarDesdeArriba y el elemento sintáctico que no está asociado a un valor de color en la paleta se puede codificar como una muestra de escape. En algunos ejemplos, como se describe en mayor detalle a continuación, el elemento sintáctico puede asociarse a un índice que no está asociado a un valor de color en la paleta. En algunos ejemplos, los valores de los píxeles codificados como muestras de escape pueden señalizarse después de la racha de índices, y no es necesario que los píxeles codificados como muestras de escape sean los mismos. En otros ejemplos, una muestra de escape puede representarse mediante la modalidad CopiarDesdeArriba y una muestra de escape puede incluirse en el mismo grupo de píxeles junto con las muestras codificadas sin escape (por ejemplo, píxeles para los que se representan valores de color en la paleta), donde el grupo de píxeles se identifica por un valor de racha. De esta forma, las técnicas se pueden usar para aumentar la longitud de las rachas, lo que puede mejorar la eficacia de la codificación.

[40] 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. Tal como se utiliza en el presente documento, el término "codificador de vídeo" se refiere genéricamente tanto a codificadores de vídeo como a decodificadores de vídeo. En esta

divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o la decodificación de vídeo. 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 paleta de acuerdo a varios 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 codificación basada en paleta o bien codificación no basada en paleta. Las modalidades de codificación no basadas en paletas pueden referirse a varias modalidades de codificación temporal inter-predictiva o a modalidades de codificación espacial intra-predictiva, tales como las diversas modalidades de codificación especificadas por el borrador 10 de la HEVC.

[41] 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. En consecuencia, 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 los datos de vídeo codificados, generados por el dispositivo de origen 12. En consecuencia, el dispositivo de destino 14 puede denominarse dispositivo de decodificación de vídeo o aparato de decodificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo.

[42] 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 de mano tales como los denominados teléfonos "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, ordenadores de coche o similares.

[43] 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 desplazar 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 permiten al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. En este ejemplo, el dispositivo de origen 12 puede modular los datos de vídeo codificados de acuerdo a 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 cableados, 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 encaminadores, conmutadores, estaciones base u otros equipos que faciliten la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[44] En otro ejemplo, el canal 16 puede incluir un medio de almacenamiento que almacena datos de vídeo codificados, generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder al medio de almacenamiento, por ejemplo, mediante el acceso a disco o el acceso a 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.

[45] 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 u otro dispositivo de almacenamiento intermedio mediante flujo de transmisión o descarga. El servidor de ficheros 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 ficheros se incluyen servidores de la Red (por ejemplo, para una sede de la Red), servidores del protocolo de transferencia de ficheros (FTP), dispositivos de almacenamiento conectados a red (NAS) y unidades de disco local.

[46] 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 de Wi-Fi), 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 ficheros puede ser una transmisión por flujo, una transmisión de descarga o una combinación de ambas.

[47] Las técnicas de esta divulgación no están limitadas a aplicaciones o configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo, en soporte de una diversidad de aplicaciones multimedia, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo por flujo, 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 prestar soporte a la transmisión de vídeo unidireccional o bidireccional para dar soporte a aplicaciones tales como transmisión por flujo de vídeo, reproducción de vídeo, difusión de vídeo y/o videotelefonía.

5 **[48]** El sistema de codificación de vídeo 10 ilustrado en 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 de una memoria local, se transmiten por flujo a través de una red o similar. Un dispositivo de codificación de vídeo puede codificar y almacenar datos en la memoria, y/o un dispositivo de decodificación de vídeo puede recuperar y decodificar datos desde la memoria. En muchos ejemplos, la codificación y 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.

15 **[49]** 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. El origen de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una videocámara, un fichero de vídeo que contiene 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 dichos orígenes de datos de vídeo.

20 **[50]** El codificador de vídeo 20 puede codificar datos de vídeo procedentes del origen de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 directamente transmite 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 un servidor de ficheros para un acceso posterior mediante el dispositivo de destino 14 para su decodificación y/o reproducción.

30 **[51]** 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 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.

35 **[52]** 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), conjuntos de compuertas programables sobre el terreno (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. Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador (CÓDEC) combinado en un dispositivo respectivo.

50 **[53]** Esta divulgación puede referirse en general al codificador de vídeo 20 que "señaliza" o "transmite" cierta información a otro dispositivo, tal como el decodificador de vídeo 30. El término "señalizar" 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 ocurrir 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 a continuación un dispositivo de decodificación puede recuperar en cualquier momento tras haber sido almacenado en este medio.

60 **[54]** En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 funcionan de acuerdo a 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 de visualización múltiple y codificación tridimensional 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 a otras normas de codificación de vídeo, tales como la norma ITU-TH.264 / AVC o normas futuras. En consecuencia, 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.

- 5 **[55]** En la HEVC y 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 formaciones de muestras, denominadas S_L , S_{Cb} y S_{Cr} . S_L es una formación bidimensional (es decir, un bloque) de muestras de luma. S_{Cb} es una formación bidimensional de muestras de crominancia Cb. S_{Cr} es una formación bidimensional de muestras de crominancia Cr. Las muestras de crominancia también se pueden denominar en el presente documento muestras de "croma". En otros casos, una imagen puede ser monocromática y puede solo incluir una formación de muestras de luma.
- 10 **[56]** Para generar una representación codificada de una imagen, el codificador de vídeo 20 puede generar un conjunto de unidades de árbol de codificación (CTU). Cada una de las CTU puede ser un bloque de árbol de codificación de muestras de luma, dos correspondientes bloques de árbol de codificación de muestras de croma y estructuras sintácticas usadas para codificar las muestras de los bloques de árbol de codificación. Un bloque de árbol de codificación puede ser un bloque de muestras de tamaño $N \times N$. Una CTU también puede denominarse
- 15 "bloque de árbol" o una "máxima unidad de codificación" (LCU). Las CTU de la HEVC pueden ser análogas en gran medida a los macrobloques de otras normas, tales como la 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 cuadrilado.
- 20 **[57]** Para generar una CTU codificada, el codificador de vídeo 20 puede llevar a cabo 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 $N \times N$. Una CU puede ser un bloque de codificación de muestras de luma y dos bloques de codificación correspondientes de muestras de croma de una imagen que tiene una formación
- 25 de muestras de luma, 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 aplica la misma predicción. Una unidad de predicción (PU) de una CU puede ser un bloque de predicción de muestras de luma, dos bloques de predicción correspondientes de muestras de croma de una imagen y estructuras sintácticas usadas para predecir las muestras
- 30 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.
- 35 **[58]** El codificador de vídeo 20 puede usar intra-predicción o inter-predicción para generar los bloques predictivos para una PU. Si el codificador de vídeo 20 utiliza intra-predicción 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.
- 40 **[59]** Si el codificador de vídeo 20 utiliza inter-predicción 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 una o más imágenes distintas a la imagen asociada a la PU. El codificador de vídeo 20 puede usar uni-predicción o bi-predicción para generar los bloques predictivos de una PU. Cuando el codificador de vídeo 20 usa la predicción unidimensional para generar los bloques predictivos para una PU, la PU puede tener un solo vector de movimiento (MV). Cuando el codificador de vídeo 20 usa predicción bidireccional para generar los bloques predictivos para una
- 45 PU, la PU puede tener dos MV.
- [60]** 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
- 50 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 de Cr predictivos de la CU y una muestra correspondiente en el bloque de codificación de Cr original de la CU.
- 55 **[61]** 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 aplica 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 utilizadas 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
- 60 bloque de transformación de luma asociado a la TU puede ser un sub-bloque del bloque residual de luma de la CU.
- 65

El bloque de transformación de Cb puede ser un sub-bloque del bloque residual de Cb de la CU. El bloque de transformación de Cr puede ser un sub-bloque del bloque residual de Cr de la CU.

[62] 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 transformación puede ser una cantidad escalar. El codificador de vídeo 20 puede aplicar una o más transformaciones a un bloque de transformación 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 transformaciones a un bloque de transformación de Cr de una TU para generar un bloque de coeficientes de Cr para la TU.

[63] 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 indican los coeficientes de transformación cuantizados. Por ejemplo, el codificador de vídeo 20 puede realizar la Codificación Aritmética Binaria Adaptativa al Contexto (CABAC) en los elementos sintácticos que indican los coeficientes de transformación cuantizados.

[64] Con respecto a la CABAC, como ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden seleccionar un modelo de probabilidad (también denominado modelo de contexto) para codificar símbolos asociados a un bloque de datos de vídeo basándose en el contexto. Por ejemplo, un modelo de contexto (*Ctx*) puede ser un índice o desplazamiento que se aplica para seleccionar uno entre una pluralidad de contextos diferentes, cada uno de los cuales puede corresponder a un modelo de probabilidad particular. En consecuencia, habitualmente se define un modelo de probabilidad diferente para cada contexto. Después de codificar o decodificar el contenedor, el modelo de probabilidad se actualiza adicionalmente basándose en el valor del contenedor para reflejar las estimaciones de probabilidad más recientes para el contenedor. Por ejemplo, un modelo de probabilidad puede mantenerse como un estado en una máquina de estados finitos. Cada estado particular puede corresponder a un valor de probabilidad específico. El siguiente estado, que corresponde a una actualización del modelo de probabilidad, puede depender del valor del contenedor actual (por ejemplo, el contenedor que se está codificando actualmente). Por consiguiente, la selección de un modelo de probabilidad puede estar influenciada por los valores de los contenedores previamente codificados, porque los valores indican, al menos en parte, la probabilidad de que el contenedor tenga un valor dado. El proceso de codificación del contexto, descrito anteriormente, puede denominarse generalmente una modalidad de codificación adaptativa al contexto.

[65] Por lo tanto, el codificador de vídeo 20 puede codificar un símbolo de destino usando un modelo de probabilidad. Asimismo, el decodificador de vídeo 30 puede analizar un símbolo de destino usando el modelo de probabilidad. En algunos casos, el codificador de vídeo 20 puede codificar elementos sintácticos usando una combinación de codificación adaptativa al contexto y no adaptativa al contexto. Por ejemplo, el codificador de vídeo 20 puede codificar por contexto contenedores seleccionando un modelo de probabilidad o "modelo de contexto" que opera sobre el contexto para codificar los contenedores. Por el contrario, el codificador de vídeo 20 puede codificar por omisión los contenedores eludiendo u omitiendo el proceso de codificación aritmética regular cuando se codifican los contenedores. En dichos ejemplos, el codificador de vídeo 20 puede usar un modelo de probabilidad fijo para codificar por omisión los contenedores. Es decir, los contenedores codificados por omisión no incluyen actualizaciones de contexto o probabilidad.

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

[67] Diferentes tipos de unidades de NAL pueden encapsular diferentes tipos de las 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).

[68] 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. Por ejemplo, el decodificador de vídeo 30 puede usar los MV de las PU para determinar bloques predictivos 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 transformaciones inversas en los bloques de coeficientes de transformación para reconstruir los bloques de transformación 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 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.

[69] La HEVC contiene varias propuestas para hacer que el códec sea más compatible con el paralelismo, incluyendo las teselas y el procesamiento paralelo de frentes de onda (WPP). El borrador de trabajo 10 de la HEVC define las teselas como un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en un escaneo cuadrado de bloques de árbol de codificación de la tesela. La división de cada imagen en teselas es una partición. El número de teselas y la ubicación de sus límites se pueden definir para toda la secuencia o cambiar de una imagen a otra. Los límites, de manera similar a los límites de fragmento, rompen las dependencias de análisis sintáctico y de predicción, de modo que una tesela se pueda procesar de forma independiente, pero los filtros en bucle (desbloqueo y desplazamiento adaptativo de muestra (SAO)) aún pueden cruzar los límites de las teselas. El borrador de trabajo 10 de la HEVC también especifica algunas restricciones en la relación entre los fragmentos y las teselas.

[70] 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 intra-predictiva o inter-predictiva, 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.

[71] 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.

[72] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para construir una paleta usando entradas de paleta de uno o más bloques codificados previamente. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para copiar una o más entradas de paleta para codificar un bloque actual, a partir de una paleta predictor que incluye entradas de paleta de bloques codificados previamente usando la modalidad de paleta u otras muestras reconstruidas. Para cada entrada en la paleta predictor, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar un indicador binario para indicar si la entrada asociada al indicador se copia a la paleta actual (por ejemplo, indicado por indicador = 1). La cadena de indicadores binarios se puede denominar vector de predicción de paleta binaria. La paleta para codificar un bloque actual también puede incluir un cierto número de nuevas entradas de paleta, que pueden codificarse explícitamente (por ejemplo, por separado del vector de predicción de paleta).

[73] Como se ha indicado anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar un cierto número de modalidades diferentes de codificación de paleta para codificar índices de una paleta. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar una modalidad de Escape, una modalidad de CopiarDesdeArriba (también denominada modalidad de Copiar-Arriba), o una modalidad de Valor (también denominado modalidad de Índice) para codificar índices de un bloque. En general, codificar una muestra usando la "modalidad de Escape" puede referirse en general a la codificación de una muestra de un bloque que no tiene un color correspondiente representado en una paleta para codificar el bloque. Como se ha indicado anteriormente, tales muestras pueden denominarse muestras de escape o píxeles de escape.

[74] Como otro ejemplo, en un tercer experimento central de codificación de contenido de pantalla, la sub-prueba B.6, como se describe en el artículo de Yu-Wen Huang y otros, "Description of Screen Content Core Experiment 3 (SCCE3): Palette Mode," ["Descripción del experimento central de contenido de pantalla 3 (SCCE3): Modalidad de Paleta,"] JCTVC-Q1123, Valencia, ES, del 27 de marzo al 4 de abril de 2014 (en adelante, Q1123), se introdujo otra modalidad en el software publicado por Canon el 26 de mayo de 2014. La macro para esta modalidad se puede definir como "CANON_NUEVA_RACHA_ÚLTIMA_TRANSICIÓN" y se puede denominar en este

documento modalidad de Racha de Transición. La Racha de Transición puede ser similar a la modalidad de Valor en cuanto a que el codificador de vídeo 20 o el decodificador de vídeo 30 puede codificar un valor de índice seguido por una racha que especifica el número de muestras posteriores que tienen el mismo índice de paleta.

5 **[75]** Una diferencia entre la modalidad de Valor y la modalidad de Racha de Transición es que el valor de índice de la modalidad de racha de transición no se señala en el flujo de bits. En cambio, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden deducir el valor de índice para la modalidad de Racha de Transición. Como se describe en el presente documento, deducir un valor puede referirse a la determinación de un valor sin referencia a la sintaxis dedicada que representa el valor que se codifica en un flujo de bits. Es decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden deducir un valor sin codificar un elemento sintáctico dedicado para el valor en un flujo de bits. El índice deducido se puede denominar índice de transición.

15 **[76]** En algunos ejemplos, puede haber dos formas de señalar las modalidades de paleta. Una primera técnica para señalar modalidades de paleta se puede denominar señalización de escape explícita. Por ejemplo, en el documento JCTVC-Q0094, si la macro "PLT_ELIMINAR_INDICADOR_ESCAPE" es cero, el codificador de vídeo 20 puede codificar explícitamente un indicador de escape para cada muestra de un bloque, para indicar si una muestra que está siendo codificada en un bloque está codificada en modalidad de Escape. Si la muestra no está codificada con la modalidad de Escape, el codificador de vídeo 20 puede codificar datos adicionales para indicar si la modalidad es CopiarDesdeArriba o Valor. En algunos casos, los datos adicionales pueden ser un indicador, denominado en este documento indicador SPoint (por ejemplo, un valor de cero del indicador SPoint puede indicar la modalidad de CopiarDesdeArriba y un valor de uno del indicador SPoint puede indicar la modalidad de Valor, o viceversa).

25 **[77]** Por lo tanto, con la señalización de escape explícita, el indicador SPoint puede usarse para indicar un tipo de racha particular para una racha de valores de pixel asociados a la modalidad indicada. Por ejemplo, el codificador de vídeo 20 puede codificar un indicador SPoint para indicar si el índice que se está codificando actualmente, y la racha de los valores de índice subsiguientes que se codifican en una racha, se codifican utilizando la modalidad de CopiarDesdeArriba o la modalidad de Valor. El codificador de vídeo 20 no codifica el indicador de escape (por ejemplo, "PLT_ELIMINAR_INDICADOR_ESCAPE") y el indicador SPoint (cuando es necesario) para las muestras de rachas posteriores. Es decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden deducir los valores del indicador de escape y del indicador SPoint para las muestras incluidas en una racha. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar el valor del indicador de escape y del indicador SPoint para las muestras incluidas en la racha sin referencia a la sintaxis dedicada que representa dichos valores en el flujo de bits.

35 **[78]** Una segunda técnica para señalar modalidades de paleta se puede denominar señalización de escape implícita. Por ejemplo, si la macro "PLT_ELIMINAR_INDICADOR_ESCAPE" del documento JCTVC-Q0094 es uno, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aumentar en uno el número de entradas de paleta de una paleta, para asimilar un índice especial para la paleta que no corresponde a ninguno de los índices del bloque. En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden incluir el índice adicional como el último índice de paleta en la paleta incrementada para un bloque dado. El índice adicional se puede usar como una indicación de una muestra de escape.

45 **[79]** En esta señalización de escape implícita, el codificador de vídeo 20 puede codificar, para un valor de muestra particular de un bloque, datos que representan el índice adicional para indicar que la muestra adicional está codificada usando la modalidad de Escape. El codificador de vídeo 20 también puede codificar el valor de color de la muestra de escape. En consecuencia, en el caso de la señalización de escape implícita, solo hay dos modalidades posibles (por ejemplo, la modalidad de CopiarDesdeArriba o la modalidad de Valor) a señalar utilizando la sintaxis explícita. Por ejemplo, solo se puede señalar el indicador SPoint para distinguir entre las modalidades. Si una muestra está codificada en la modalidad de Valor y el índice para la modalidad de Valor es igual al índice de escape (por ejemplo, el índice adicional señalado anteriormente para la paleta), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden deducir que la muestra está codificada en modalidad de Escape. En este caso, no se señala ninguna racha. Cuando se utiliza la señalización de escape implícita con la modalidad de Racha de Transición, el indicador SPoint puede tomar valores 0 (por ejemplo, modalidad de Valor), 1 (por ejemplo, modalidad de CopiarDesdeArriba) o 2 (por ejemplo, modalidad de Racha de Transición).

60 **[80]** Las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de una o más entre predicción de paleta y señalización de índices de paleta con señalización de escape implícita. Por ejemplo, aspectos de esta divulgación se relacionan con reiniciar una paleta predictora. Por ejemplo, reiniciar una paleta predictora puede incluir fijar el tamaño de la paleta predictora en cero, de modo que la paleta predictora no se pueda usar para construir una paleta para un bloque que se está codificando actualmente. En este caso, un vector de predicción de paleta binaria no se señala en un flujo de bits y no se predicen entradas de paleta a partir de la paleta predictora.

65 **[81]** En el documento JCTVC-Q0094, una paleta predictora se reinicia para un bloque de árbol de codificación (CTB) en el borde izquierdo de la imagen (por ejemplo, al comienzo de una fila de los CTB en una imagen ("fila de

CTB")). El criterio racional para reiniciar la paleta predictora al comienzo de una fila de CTB puede ser doble. En primer lugar, después de codificar una línea de los CTB, la paleta predictora puede incluir entradas de paleta que en su mayoría pertenecen a paletas de bloques ubicados en el lado derecho de la imagen. En consecuencia, la paleta predictora puede no ser muy efectiva para predecir una paleta actual para un primer CTB de una fila de CTB, que puede estar situada en el lado izquierdo de la imagen (por ejemplo, los colores de los píxeles en el lado izquierdo de una imagen pueden diferir de los colores de los píxeles en el lado derecho de la imagen). Además, cuando se utilizan dos o más frentes de onda para codificar una imagen particular, además de los estados de la CABAC, sería necesario propagar información sobre la paleta predictora entre frentes de onda.

[82] Por ejemplo, un frente de onda puede ser una unidad de datos de vídeo para su codificación, y puede codificarse en paralelo con otros frentes de onda. Por ejemplo, dos enfoques de paralelización se incluyen en la HEVC, incluyendo el procesamiento en paralelo de frente de onda (WPP) y el frente de onda superpuesto (OWF). El WPP permite crear particiones de imágenes que pueden procesarse en paralelo sin incurrir en grandes pérdidas de codificación. En el WPP, las filas de los CTB se procesan en paralelo a la vez que se conservan todas las dependencias de codificación. El OWF admite la superposición de la ejecución de imágenes consecutivas utilizando frentes de onda. En el documento P0114 por Laroche G; Poirier T; Gisquet C; Onno, la paleta predictora se reinicializa solo al comienzo de una nueva línea de CTB. Además, solo la paleta de la última CU de Paleta decodificada se usa como predictora.

[83] Sin embargo, en algunos casos, reiniciar una paleta predictora puede conducir a pérdidas de codificación. Por ejemplo, sin una paleta predictora, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos que indican todas las entradas de una paleta (por ejemplo, todos los índices de paleta y valores de color relacionados) en un flujo de bits. Estos datos de paleta pueden ser una cantidad relativamente grande de datos con respecto a las entradas de paleta predichas, que pueden indicarse en el flujo de bits usando un solo indicador. En consecuencia, reiniciar una paleta predictora puede afectar negativamente a la tasa de bits de los datos de vídeo codificados con codificación basada en paletas.

[84] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para inhabilitar el reinicio de una paleta predictora basándose en que el número de frentes de onda que se usan para codificar una imagen sea uno o cero (es decir, sin frentes de onda). Es decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no reiniciar una paleta predictora cuando se usan uno o cero frentes de onda para codificar una imagen. Sin embargo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para permitir el reinicio de una paleta predictora cuando se usan dos o más frentes de onda para codificar una imagen. En este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden no necesitar propagar información acerca de una paleta predictora entre frentes de onda, evitando al mismo tiempo pérdidas de codificación en el caso de frentes únicos de onda.

[85] Según aspectos de esta divulgación, cuando se usan múltiples teselas y múltiples frentes de onda para codificar una imagen (por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 usan dos o más teselas y dos o más frentes de onda para codificar la imagen), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reiniciar la paleta predictora al comienzo de cada fila de CTB de teselas. Una fila de CTB de teselas puede incluir los CTB de una fila de CTB que pertenecen a una tesela en particular. Debería entenderse que una "fila" puede estar orientada verticalmente u horizontalmente, en función de un orden de escaneo particular usado para codificar bloques. Una fila también se puede denominar línea en el presente documento, y los términos se pueden usar de forma intercambiable en este documento.

[86] Las técnicas descritas anteriormente también se pueden aplicar a una tabla de transición de paleta. Por ejemplo, el documento de presentación estándar de Yu-Chen Sun y otros, "CE6 Test C.2 Transition Copy Mode," ["Modalidad de copia de transición CE6 Prueba C.2,"] JCTVC-S0078, Estrasburgo, FR, 17 al 24 de octubre (JCTVC-S0078) describe una modalidad de copia de transición. La copia de transición también se puede denominar modalidad de racha de copia de transición (racha de TC). En la modalidad de racha de TC, un decodificador registra los patrones de índice de color adyacentes y utiliza los patrones para predecir los índices a codificar. En la modalidad de racha de TC, como ejemplo, un píxel actual, C, tiene un índice de color piloto igual a uno. Desde un área causal de píxeles codificados, un codificador de vídeo (tal como el codificador de vídeo 20 o el decodificador de vídeo 30) puede determinar que un índice de color, a la derecha de un índice de color que sea igual a uno, probablemente sea dos (por ejemplo, suponiendo un escaneo de izquierda a derecha). Este patrón de transición de uno a dos puede registrarse en una tabla de copia de transición (tabla de TC) para mejorar la eficacia de predicción del índice de color.

[87] En el documento JCTVC-S0078, una tabla de TC se reinicia cuando una CU actual que se está codificando es la primera CU del fragmento, o la CU actual hereda la tabla de TC de la CU anterior cuando la CU actual es una CU no primera del fragmento. Esta propiedad de herencia se puede denominar "propagación de la tabla de TC". La tabla de TC puede mantenerse y actualizarse a medida que se codifican los píxeles en la CU.

[88] En algunos ejemplos, en la modalidad de copia de transición, se mantienen dos tablas de transición, una para filas de índices pares y la otra para filas de índices impares. Contribuciones anteriores describieron el uso de

una tabla de transición única. En cualquier caso, en general, como se ha indicado anteriormente, se puede usar una tabla de transición de una CU (o bloque) codificada de una paleta anterior para inicializar la tabla de transición para una CU (o bloque) actual codificada en la modalidad de paleta.

5 **[89]** De acuerdo a aspectos de esta divulgación, el concepto de reiniciar la paleta al comienzo de un primer CTB en una fila de CTB o el primer CTB en una tesela en una fila de CTB cuando se utilizan múltiples frentes de onda, como se ha descrito anteriormente, puede extenderse a reiniciar asimismo la tabla (o tablas) de transición de paleta en esas ubicaciones. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para fijar todas las entradas en una tabla de transición de paleta en cero (o en cualquier otro índice fijo de paleta, o cualquier otro patrón) para el primer CTB en una fila de CTB o el primer CTB en una tesela en una fila de CTB cuando se utilizan múltiples frentes de onda (o, de manera equivalente, las primeras CU codificadas por paleta dentro de los CTB). De acuerdo a aspectos de esta divulgación, el reinicio de la paleta predictora y de las tablas de transición de paleta puede generalizarse a cualquier otro dato relacionado con paletas que pueda propagarse desde una CU codificada por paleta a otra.

15 **[90]** En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar un indicador de un conjunto de parámetros de vídeo (VPS), un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS), una cabecera de fragmento, a nivel de tesela, a nivel de bloque o en otro lugar, que indica si se realiza un reinicio de paleta predictora (y / o reinicio de otra información tal como las tablas de transición de paletas mencionadas anteriormente) al comienzo de cada fila de CTB. En los casos en que se señala el indicador a nivel de tesela, el indicador puede indicar si el reinicio de la paleta predictora se realiza al comienzo de cada fila de CTB de teselas (por ejemplo, al comienzo de cada fila de los CTB en una tesela). Alternativamente, para el primer bloque codificado usando la modalidad de paleta en una fila de los CTB, se puede señalar un indicador para indicar si se realiza el reinicio de la paleta predictora.

20 **[91]** En otro ejemplo, se puede señalar un indicador (por ejemplo, donde un indicador puede ser uno o más elementos sintácticos) para cada bloque codificado usando la modalidad de paleta para indicar si se debería realizar el reinicio de la paleta predictora. En algunos ejemplos, esta técnica puede tener la ventaja de permitir que el codificador de vídeo 20 y el decodificador de vídeo 30 puedan seleccionar entre técnicas que tienen todas las entradas de paleta explícitamente señalizadas o técnicas que usan una paleta predictora para predecir algunas de las entradas de paleta. En algunos ejemplos, el valor del indicador puede deducirse. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir el valor del indicador para cada frente de onda, de manera que el valor deducido indique que la paleta predictora está reiniciada. Como alternativa, o adicionalmente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir el valor del indicador para el primer bloque de un fragmento o tesela, de modo que el valor deducido indique que una paleta predictora está reiniciada al comienzo de cada fragmento y / o tesela.

25 **[92]** En otro ejemplo, en vez de señalar un indicador para indicar un reinicio de paleta predictora, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reiniciar una paleta predictora según una regla o conjunto de reglas. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aplicar la regla o conjunto de reglas. En un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aplicar la siguiente regla para determinar si se reinicia una paleta predictora: si la paleta en el bloque actual no tiene entradas predichas desde una paleta predictora, la paleta predictora se reinicia (por ejemplo, la paleta predictora se fija para que tenga una longitud cero) para este bloque actual, de modo que solo la paleta actual pueda usarse como predictora para los siguientes bloques, pero no las entradas de paleta de los bloques previos.

30 **[93]** En el ejemplo anterior, el codificador de vídeo 20 puede controlar el reinicio de paleta. Por ejemplo, cuando se ha de reiniciar la paleta, el codificador de vídeo 20 simplemente no puede usar ninguna entrada de la paleta predictora. Una ventaja potencial de implementar este proceso es que no se agregan nuevos elementos sintácticos, y el reinicio de paleta puede ser más granular y más flexiblemente controlado. Por ejemplo, la flexibilidad / granularidad se puede lograr debido a la capacidad de cambiar las condiciones en las que se realiza el reinicio de la paleta predictora.

35 **[94]** En algunos ejemplos, el codificador de vídeo 20 no puede usar la paleta predictora en al menos uno de los siguientes casos: para la primera LCU en cada fila de LCU, cuando la codificación de frentes de onda está habilitada; para el primer bloque en una tesela o para el primer bloque en un fragmento.

40 **[95]** De acuerdo a aspectos de esta divulgación, se puede reinicializar una paleta predictora. Por ejemplo, la reinicialización de la paleta predictora generalmente se puede referir a la sustitución de entradas de la paleta predictora por nuevas entradas. Por ejemplo, después de codificar un bloque de índices, una paleta predictora puede incluir un cierto número de entradas para predecir una paleta del siguiente bloque a codificar. La reinicialización de la paleta predictora puede incluir la eliminación del número de entradas y la adición de nuevas entradas a la paleta predictora para predecir la paleta del próximo bloque a codificar.

65

[96] En algunos ejemplos, según aspectos de esta divulgación, al comienzo de cada línea de CTB, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar una paleta predictora para determinar una paleta actual de un CTB actual usando entradas de paleta de un primer CTB de la línea de CTB encima del CTB actual, si la línea de CTB actual no es la primera línea de CTB en la imagen / fragmento / tesela. En este caso, como se describe con mayor detalle con respecto al ejemplo de la FIG. 5 a continuación, después de terminar la codificación del primer CTB y antes de codificar el segundo CTB en una línea de CTB arriba (siempre que el primer CTB no sea la primera línea de CTB en la imagen / fragmento / tesela), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar la paleta predictora para reinicializar la paleta predictora al comienzo de la línea de CTB actual. Si la línea actual de CTB es la primera en la imagen / fragmento / tesela, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para fijar el número de entradas de paleta predictora (por ejemplo, como se describe en el documento JCTVC-Q0094) y no predecir ninguna entrada de paleta al determinar la paleta para el CTB actual.

[97] En otro ejemplo, el concepto de reinicializar una paleta predictora puede generalizarse. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para generar una paleta predictora después de la codificación de los N CTB de una línea de los CTB por encima de un CTB que está siendo codificado actualmente y antes de comenzar a codificar el $N+1^{\text{ésimo}}$ CTB en la línea de CTB por encima del CTB que está siendo codificado actualmente (siempre que la línea de los CTB no sea la primera línea de CTB en la imagen / fragmento / tesela). El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar la paleta predictora para determinar la paleta del CTB que se está codificando actualmente utilizando la paleta predictora generada. El CTB que se está codificando actualmente puede ser el primer CTB de una línea de CTB actual (siempre que el CTB actual no sea la primera línea de CTB en la imagen / fragmento / tesela). Aquí, N puede estar entre uno y el número de los CTB en la línea de CTB. Para alinear conceptualmente esta técnica de reinicialización de paleta predictora con la codificación de frentes de onda, se puede seleccionar N para que sea menor o igual a dos. De esta forma, se puede usar el mismo proceso de reinicialización de paleta predictora, independientemente del número de frentes de onda utilizados para codificar una imagen.

[98] Como se ha indicado anteriormente, un CTB también puede denominarse una LCU. En algunos ejemplos, la reinicialización de paleta predictora puede realizarse para el primer CTB de una línea de bloques en la que se usa una modalidad de paleta para predecir cualquier bloque del CTB, por ejemplo, en casos en los que se pueden usar varias modalidades de predicción (inter-modalidades o intra-modalidades) o de paleta para codificar un CTB. Si bien los ejemplos anteriores se describen con respecto a los CTB con fines de ilustración, debería entenderse que las técnicas se pueden extender a cualquier tipo de bloque de datos de vídeo.

[99] De acuerdo a aspectos de esta divulgación, el proceso de reinicialización de paleta descrito anteriormente puede extenderse a tablas de transición de paleta. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar tablas de transición paleta en el comienzo de una fila actual de CTB utilizando las tablas de transición de paleta generadas después de la codificación de los N CTB y antes de comenzar a codificar el $N+1^{\text{ésimo}}$ CTB en la línea de CTB por encima del CTB actual (siempre que el CTB actual no sea la primera línea de CTB en la imagen / fragmento / tesela). Aquí, N puede estar entre uno y el número de los CTB en la línea de CTB. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar un valor de N que sea menor o igual a dos.

[100] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar para copiar el tamaño de las tablas de transición además de las mismas tablas de transición de paleta. De hecho, este proceso de reinicialización de información de codificación de paleta (por ejemplo, información de paleta predictora, información de tabla de transición de paleta o similar) puede generalizarse a cualquier otro dato relacionado con paleta que se propague desde una CU codificada por paleta a la siguiente CU codificada por paleta. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar datos relacionados con paleta para un CTB actual basándose en datos de paleta generados después de codificar los N CTB y antes de comenzar a codificar el $N+1^{\text{ésimo}}$ CTB en la línea de CTB por encima del CTB actual.

[101] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar la reinicialización para todos los CTB aplicables (por ejemplo, los CTB inicialmente codificados en una línea de CTB). En otros ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para que solo puedan realizar las técnicas de reinicialización cuando el número de frentes de onda que se usan para codificar una imagen o tesela sea dos o más.

[102] Según aspectos de esta divulgación, cuando las teselas y los frentes de onda se usan juntos para codificar una imagen particular, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aplicar las técnicas de reinicialización a las teselas como si las teselas fueran imágenes independientes. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aplicar el proceso de reinicialización descrito anteriormente para un primer CTB de tesela si el primer CTB de tesela no es la primera línea de CTB en la tesela, como se describe en mayor detalle con respecto a la FIG. 7 a continuación.

[103] Otros aspectos de esta divulgación se refieren a muestras de escape de señalización en ejemplos en los que se usa señalización de escape implícita. Por ejemplo, en el documento JCTVC-Q0094, para la técnica de escape implícita descrita anteriormente, el tamaño de la paleta se incrementa en uno y el último índice en la paleta expandida se asigna para indicar si una muestra está codificada como una muestra de escape. Si se utiliza codificación binaria truncada para codificar los índices, el último índice puede usar un bit más que el índice cero. De forma similar, si se usa otra estrategia de codificación de longitud variable, la diferencia en las longitudes puede ser incluso mayor.

[104] Normalmente, para ciertos tamaños de bloques, puede haber más muestras que están codificadas como muestras de escape que bloques de otros tamaños. Para aprovechar esta relación para mejorar la codificación de índices, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para asignar el índice cero de una paleta al índice usado para indicar una muestra de escape para bloques que probablemente tengan más píxeles codificados como una muestra de escape. Para los otros bloques que pueden tener relativamente menos muestras codificadas como muestras de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar para que puedan seguir el esquema actual de asignación del índice más alto de la paleta expandida para indicar una muestra de escape.

[105] En un ejemplo con fines de ilustración, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para asignar un índice cero al índice usado para indicar una muestra de escape para los tamaños de bloque de 32×32 y menores. En este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para asignar el índice más alto (por ejemplo, el más grande numéricamente de los índices de paleta) al índice usado para indicar una muestra de escape para bloques de tamaño 64×64 y mayores.

[106] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para implementar la misma regla de asignación de índice de escape para cada tamaño de bloque a priori. En otro ejemplo, una regla de asignación de índice de escape puede señalizarse explícitamente en un conjunto de parámetros de vídeo (VPS), un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS), una cabecera de fragmento, a nivel de mosaico o en otro lugar.

[107] Otros aspectos de esta divulgación se refieren a muestras de escape de señalización en ejemplos en los que se usa señalización de escape explícita. En el caso de señalización de escape explícita, la modalidad de escape se puede señalar con un indicador. De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar de manera adaptativa al contexto los indicadores de escape basándose en el tamaño del bloque. Por ejemplo, los contextos pueden diseñarse para codificar indicadores de escape con el fin de explotar la relación de que, para ciertos tamaños de bloques, puede haber más muestras codificadas usando una muestra de escape con respecto a otros tamaños de bloques. En este caso, el contexto para el indicador puede depender del tamaño del bloque. Además, en algunos ejemplos, un tamaño de paleta puede tener cierta correlación con la cantidad / el número de muestras de escape a señalar en la CU. Por lo tanto, el contexto puede depender del tamaño de la paleta o de si el tamaño de la paleta es igual al tamaño máximo de paleta (o no igual al tamaño máximo de paleta).

[108] Otros aspectos de esta divulgación se refieren a la señalización de valores de racha para rachas de índices codificados usando la modalidad de paleta de Valor. Por ejemplo, en el documento estándar de presentación de Guillaume Laroche et al., "AHG10: Run Coding for Palette Mode" [Codificación de Rachas para Modalidad de Paleta], JCTVC-Q0066, Valencia, ES, 27 de marzo a 4 de abril de 2014 (JCTVC-Q0066), se propone que para cada bloque codificado utilizando una paleta se pueda señalar un umbral de índice. Si el índice que se codifica en la modalidad de Valor es mayor o igual que el umbral del índice, no se señala ninguna racha para la modalidad de Valor. Esta divulgación puede referirse a este procedimiento como "racha limitada".

[109] De acuerdo a aspectos de esta divulgación, se pueden hacer modificaciones al proceso descrito en el documento Q0066. En un primer ejemplo, los primeros índices indicados por el elemento sintáctico `núm_salto`, siempre se señala una racha. En algunos ejemplos, el elemento sintáctico `núm_salto` es igual a dos, por lo que la racha siempre se señala para los índices cero y uno. El valor indicado por el elemento sintáctico `núm_salto` puede ser conocido tanto por el codificador de vídeo 20 como por el decodificador de vídeo 30 a priori (por ejemplo, predeterminado en los codificadores de vídeo), o puede ser señalado explícitamente en el conjunto de parámetros de vídeo (VPS), el conjunto de parámetros de secuencia (SPS), el conjunto de parámetros de imagen (PPS), la cabecera de fragmento, a nivel de tesela o en otro lugar.

[110] En un segundo ejemplo, si el tamaño de la paleta está por debajo de un umbral de tamaño de paleta en particular, el umbral de índice no se señala y la racha se señala para cada índice. El umbral de tamaño de paleta puede ser conocido tanto por el codificador de vídeo 20 como por el decodificador de vídeo 30 a priori o puede ser señalado explícitamente en el conjunto de parámetros de vídeo (VPS), el conjunto de parámetros de secuencia (SPS), el conjunto de parámetros de imagen (PPS), a nivel de tesela o en otro lugar.

[111] En un tercer ejemplo, se puede señalar un indicador para cada bloque donde el tamaño de la paleta es mayor que el umbral del tamaño de la paleta, para indicar si se señala el umbral del índice. El indicador puede ser

señalizado solo para ciertos tamaños de bloque y / o tamaños de paleta. Si el umbral del índice no se señala, la racha se señala para todos los índices. En un ejemplo alternativo, en lugar de señalar un indicador explícito, se señala un umbral de índice de un tamaño de paleta para indicar que la racha se señala para todos los índices.

5 **[112]** En un cuarto ejemplo, la binarización truncada (por ejemplo, la codificación binaria truncada como se describe en la Solicitud Provisoria Estadounidense N° 62 / 002.054, presentada el 22 de mayo de 2014) puede usarse para codificar el umbral del índice para un bloque si el tamaño de la paleta está por encima del umbral de tamaño de la paleta. Si se señala un indicador explícito, como en el tercer ejemplo anterior, el valor máximo para la binarización truncada puede ser (tamaño de la paleta actual - núm_salto). Si no se envía ningún indicador explícito, el valor máximo para la binarización truncada puede ser (tamaño de la paleta actual núm_salto + 1).

15 **[113]** En un quinto ejemplo, un valor de racha puede codificarse después de reorganizar, reordenar o re-correlacionar el índice de paleta. Por ejemplo, algunas técnicas de codificación de paleta se pueden usar para agrupar índices para tener rachas más largas de índices. En un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para realizar un escaneo adaptativo de índices o aplicar una transformación de Burrows Wheeler.

20 **[114]** En un sexto ejemplo, el umbral de índice puede señalarse como (tamaño de paleta - umbral de índice) en casos en los que se señala el indicador explícito descrito anteriormente con respecto al tercer ejemplo. De forma similar, el umbral del índice se puede señalar como (tamaño de la paleta + 1 - umbral del índice) en casos en los que dicho indicador explícito no se señala. En algunos casos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar el umbral del índice o (tamaño de la paleta - umbral del índice) en función del tamaño del bloque y / o del tamaño de la paleta. En algunos ejemplos, se puede usar una estrategia similar en casos en las que no se señala un indicador explícito.

25 **[115]** Las técnicas de los seis ejemplos descritos anteriormente pueden implementarse mediante el codificador de vídeo 20 y el decodificador de vídeo 30 individualmente o en combinación. Además, las técnicas pueden implementarse selectivamente en función del tamaño del bloque y / o del tamaño de la paleta. Tal dependencia del tamaño de bloque y / o del tamaño de paleta puede ser conocida tanto por el codificador de vídeo 20 como por el decodificador de vídeo 30 a priori, o puede señalarse explícitamente en el conjunto de parámetros de vídeo (VPS), el conjunto de parámetros de secuencia (SPS), el conjunto de parámetros de imagen (PPS), la cabecera de fragmento, a nivel de tesela o en otro lugar.

35 **[116]** A diferencia del documento JCTVC-Q0066, de acuerdo a aspectos de esta divulgación, las técnicas descritas anteriormente pueden ser normativas. Por ejemplo, esto significa que a un flujo de bits válido (por ejemplo, un flujo de bits que se puede decodificar mediante un decodificador conforme a una norma particular) no se permite tener un índice para el que no se señale la aparición de rachas en posiciones consecutivas, y codificarse usando la modalidad de Valor. Es decir, dos posiciones consecutivas que tienen el mismo valor de índice no se pueden codificar utilizando la modalidad de Valor. Esta restricción se puede tratar como una restricción normativa en un flujo de bits. Es decir, un flujo de bits conforme no deberá contener dos o más píxeles consecutivos cualesquiera, codificados con la modalidad de Valor, que tengan valores de índice idénticos para los cuales no se señala una racha. Esto permite la aplicación de todas las eliminaciones de redundancia descritas en la Solicitud Provisoria Estadounidense Núm. 62 / 002.054, presentada el 22 de mayo de 2014, así como la eliminación de los denominados valores de índices imposibles descritos en el documento de C. Gisquet et al., "AHG10 : Palette Index Coding" [Codificación del índice de paleta], JCTVC-Q0064, Valencia, ES, del 27 de marzo al 4 de abril de 2014 (JCTVC-Q0064).

50 **[117]** Si también se utiliza la modalidad de Racha de Transición, la técnica de Racha Limitada también puede extenderse a la modalidad de Racha de Transición. En este caso, si la modalidad es Valor o Racha de Transición y el índice de paleta (para la modalidad de Valor) o el índice de paleta deducido (para la modalidad de Racha de Transición) está por encima del umbral del índice, no se señala ninguna racha y se puede aplicar la misma restricción normativa de flujos de bits, como se ha descrito anteriormente.

55 **[118]** Otros aspectos de esta divulgación se refieren a la codificación de un valor de racha que indica una longitud de racha de una racha de índices de paleta. En un ejemplo, las técnicas se pueden usar para codificar un valor de racha de un índice codificado usando la modalidad de Valor.

60 **[119]** Por ejemplo, en algunos ejemplos, se pueden usar tres indicadores codificados por contexto para codificar un valor de racha en modalidad de Valor. En dichos ejemplos, los tres indicadores codificados por contexto pueden incluir un indicador mayor que cero que indica si el valor de la racha es mayor que cero, un indicador mayor que uno que indica si el valor de la racha es mayor que uno y un indicador mayor que dos que indica si el valor de la racha es mayor que dos. En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar cada uno de los tres indicadores usando un solo contexto.

65 **[120]** Sin embargo, usar un único contexto puede requerir que el codificador de vídeo 20 y el decodificador de vídeo 30 actualicen el modelo de probabilidad asociado al contexto después de codificar cada uno de los

indicadores. Además, el anterior proceso de actualización de probabilidad puede introducir demoras en el proceso de codificación. En un ejemplo con fines de ilustración, tres contenedores utilizan el mismo modelo de contexto (por ejemplo, ctx (0)) con fines de codificación adaptativa al contexto. En este ejemplo, un primer contenedor puede usar ctx (0) para determinar un modelo de probabilidad para la codificación. El valor del primer contenedor influye en el modelo de probabilidad asociado a ctx (0). En consecuencia, se realiza una actualización de probabilidad antes de codificar el segundo contenedor con ctx (0). También se realiza una actualización de probabilidad adicional antes de codificar el tercer contenedor con ctx (0). De esta forma, la actualización de probabilidad puede introducir un retraso en el ciclo de codificación.

[121] Además, la correlación entre el primer contenedor, el segundo contenedor y el tercer contenedor de un valor de racha puede ser insuficiente para garantizar el tiempo y los recursos informáticos asociados a la actualización del modelo de probabilidad. Es decir, un beneficio potencial de la codificación adaptativa al contexto es la capacidad de adaptar un modelo de probabilidad basándose en contenedores previamente codificados (dado el mismo contexto). Sin embargo, si el valor de un primer contenedor tiene poca relevancia en el valor de un contenedor posterior, puede haber poca ganancia de eficacia asociada a la actualización de probabilidad. En consecuencia, los contenedores que exhiben una baja correlación pueden no beneficiarse tanto de la codificación adaptativa al contexto como los contenedores que tienen correlaciones relativamente mayores.

[122] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para determinar un contexto para codificar un valor de racha basándose en el índice del valor de la racha. Adicionalmente o alternativamente, el codificador de vídeo 20 y el decodificador 30 pueden configurarse para determinar un contexto para codificar un valor de racha basándose en un tamaño de bloque de un bloque que se está codificando actualmente, o en un tamaño de paleta de una paleta para el bloque que se está codificando actualmente. En algunos ejemplos, los aspectos de esta divulgación incluyen el uso de contextos múltiples para codificar un valor de racha.

[123] En un ejemplo, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar siete contextos para codificar un valor de racha. Por ejemplo, se pueden definir siete contextos para codificar un indicador que indique que un valor de racha es mayor que cero. Este ejemplo se ilustra en la Tabla 1 a continuación:

Tabla 1 - SIETE CONTEXTOS PARA CODIFICAR UNA RACHA BASÁNDOSE EN EL ÍNDICE

Índice	Contexto de la racha
0	0
1	1
2	2
3	3
[4, 7]	4
[8, 15]	5
[16, 31]	6

Si bien el ejemplo de la Tabla 1 ilustra siete contextos, en otros ejemplos se puede definir un número alternativo de contextos en función de un valor de índice de la ejecución que se está codificando. La Tabla 1 puede suponer que el elemento sintáctico MÁX_TAMAÑO_PLT es 32.

[124] En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar dos contextos para codificar los valores de racha para la modalidad de Valor basándose en el valor del índice. Por ejemplo, para un índice de modalidad de Valor 0, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (0) para codificar datos que indican el valor de la racha. En este ejemplo, para todos los demás índices de Valor, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (1) para codificar datos que indican el valor de la racha.

[125] En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar dos contextos para codificar los valores de racha en la modalidad de Valor basándose en el valor del índice. En estos ejemplos, para los índices de valor 0 y 1, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (0) para codificar datos que indican el valor de la racha. Para índices de Valor mayores que 1, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (1) para codificar datos que indican el valor de la racha.

[126] En otro ejemplo más, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar tres contextos para codificar los valores de racha en la modalidad de Valor basándose en el valor del índice. Por ejemplo, para el índice de Valor 0, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (0) para codificar los datos que indican la racha. Para los índices de Valor de 1 y 2, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (1) para codificar los datos que indican la racha. Para índices de Valor mayores que 2, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (2) para codificar los datos que indican el valor de la racha.

[127] En otro ejemplo más, se pueden usar tres contextos para codificar los valores de racha en la modalidad de Valor en función del valor del índice. Por ejemplo, para los índices de valor 0 y 1, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (0) para codificar datos que indican el valor de la racha. Para índices de valor en el rango de 2 a 5, inclusive, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (1) para codificar datos que indican el valor de la racha. Para índices de Valor mayores que 5, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (2) para codificar datos que indican el valor de la racha.

[128] En otro ejemplo, se pueden usar cuatro contextos para codificar los valores de racha en la modalidad de Valor en función de (por ejemplo, basándose en) el valor del índice. Por ejemplo, para el índice de Valor 0, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (0) para codificar datos que indican el valor de la racha. Para los índices de valor 1 y 2, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (1) para codificar datos que indican el valor de la racha. Para índices de valor en el rango de 3 a 6, inclusive, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (2) para codificar datos que indican el valor de la racha. Para índices de Valor mayores que 6, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar el contexto de racha (3) para codificar datos que indican que se puede usar el valor de la racha. En algunos casos, cualquier combinación de las técnicas anteriores se puede usar para determinar el contexto.

[129] Las técnicas descritas anteriormente pueden usarse para codificar cualquier combinación de contenedores de un valor de racha binarizado. Por ejemplo, como se ha indicado anteriormente, un valor de racha binarizado puede incluir un primer contenedor que indique si el valor de racha es mayor que cero, un segundo contenedor que indique si el valor de racha es mayor que uno, un tercer contenedor que indique si el valor de racha es mayor que dos y otros contenedores necesarios cualesquiera para representar el valor de la racha. De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar los contextos definidos anteriormente para codificar el primer contenedor del valor de racha binarizado, el segundo contenedor del valor de racha binarizado, el tercer contenedor del valor de racha binarizado o cualquier combinación de los mismos. Las técnicas también se pueden aplicar a otras binarizaciones de un valor de racha.

[130] De acuerdo a aspectos de esta divulgación, el contexto puede basarse en un valor de índice real o un valor de índice ajustado. Por ejemplo, en un ejemplo, los contextos de racha pueden depender del elemento sintáctico `índice_paleta_ajustado` en lugar del elemento sintáctico `índice_paleta`, donde el elemento sintáctico `índice_paleta_ajustado` puede indicar un índice ajustado en función de las siguientes condiciones:

```
Si (índice_paleta_ajustado >= índiceRefAjustado) entonces índice_paleta =
índice_paleta_ajustado + 1
De lo contrario, (índice_paleta = índice_paleta_ajustado)
```

como se describe en el documento de Wei Pu et al., "SCCE3: Test B.12 - Binarization of Escape Sample and Palette Index" [Prueba B.12 - Binarización de muestra de escape e índice de paleta], JCTVC-R0065, Sapporo, JP, 30 de junio a 9 de julio de 2014 (JCTVC-R0065). Es decir, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para determinar el contexto para codificar un valor de racha basándose en un índice de paleta señalado en un flujo de bits (que puede estar asociado al elemento sintáctico `índice_paleta_ajustado`) en vez del índice de paleta realmente utilizado para codificar una muestra en un bloque, en casos en los que se señala un índice de paleta ajustado en el flujo de bits.

[131] En otro ejemplo, si se usa un código binario truncado para codificar el índice, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar los contextos de racha basándose en los primeros 'k' bits de la sintaxis del índice (por ejemplo, tal como se describe, para ejemplo, en la Sección 9.3.3.x del documento JCTVC-R0065). Por ejemplo, al usar código binario truncado, el elemento sintáctico para representar un índice de paleta puede usar k bits o $k+1$ bits, basándose en el valor relativo del índice de paleta para la variable u , descrita a continuación (tal como se describe en la Sección 9.3.3.x del documento JCTVC-R0065):

La entrada a este proceso es una solicitud de una binarización de un TB para un elemento sintáctico con el valor `synVal`, `cMax`. La salida de este proceso es la binarización del TB del elemento sintáctico.

Sea $n = cMax + 1$, $k = \text{piso}(\log_2(n))$ tal que $2^k \leq n < 2^{k+1}$ y sea $u = 2^{k+1} - n$.

Si $\text{synVal} < u$, la cadena de contenedores de TB se especifica mediante la representación binaria de synVal con longitud k .

De lo contrario, la cadena de contenedores de TB se especifica mediante la representación binaria de $\text{synVal} + u$ con la longitud $k + 1$.

[132] En este ejemplo, los contextos de racha solo dependen de los primeros k bits decodificados. Por lo tanto, por ejemplo, el decodificador de vídeo 30 no necesita esperar para decodificar el posible bit $k+1$ antes de determinar los contextos.

[133] En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para usar contextos múltiples basándose en el valor de índice, el tamaño de la paleta de un bloque que se está codificando actualmente y / o un tamaño del bloque que se está codificando actualmente. Si la modalidad de Racha Limitada, descrita anteriormente, se usa para codificar un bloque, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto basándose en el valor de índice, el tamaño del bloque y / o el umbral de índice por encima del cual no se señala una racha.

[134] Si también se usa la modalidad de Racha de Transición, esta técnica también puede extenderse a la modalidad de Racha de Transición. En este caso, para la modalidad de Racha de Transición, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para determinar contextos para codificar la racha en función del valor de índice, el tamaño del bloque y / o el umbral de índice por encima del cual no se señala una racha. En este caso, se pueden usar conjuntos independientes de contextos para las rachas correspondientes a la modalidad de Valor y a la modalidad de Racha de Transición.

[135] Otros aspectos de esta divulgación se refieren a la codificación de rachas de índices de paleta que se codifican usando más de una modalidad de paleta. Por ejemplo, en la modalidad CopiarDesdeArriba, descrita en el documento JCTVC-Q0094, no es posible copiar una muestra codificada como una muestra de escape o un indicador de escape desde la línea. Un ejemplo de dos filas de índices se muestra en la Tabla 2 a continuación:

TABLA 2 - ÍNDICES EJEMPLARES

Línea arriba	1	2	2	4	ESC	5	1
Línea actual	1	2	2	4	ESC	5	2

donde los valores numéricos representan índices y ESC representa una muestra codificada como una muestra de escape. En este ejemplo, comenzando desde la muestra más a la izquierda en el ejemplo anterior, según el documento JCTVC-Q0094, el codificador de vídeo 20 puede señalar al modalidad CopiarDesdeArriba. El codificador de vídeo 20 también puede indicar un valor de racha de tres, ya que no está permitido copiar valores de escape o indicadores. Por consiguiente, la muestra codificada como una muestra de escape termina la racha.

[136] El documento de presentación estándar de Jianqing Zhu et al., "AHG10: Modified Copy Above Mode for Palette Based Coding" [Modalidad de Copiar Arriba modificada para codificación basada en paleta], JCTVC-Q0174, Valencia, ES, 27 de marzo a 4 de abril de 2014 (JCTVC-Q0174) describe permitir la copia de valores de escape (por ejemplo, los valores de píxel reales) en una racha. Sin embargo, esta técnica requiere que los valores asociados a las muestras de escape sean idénticos.

[137] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar datos que indican más de una modalidad de paleta en una racha de índices. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para copiar indicadores de escape (en señalización de escape explícita) o un índice de escape (en señalización de escape implícita) con otros índices que tienen valores de color asociados en una paleta. Con respecto al ejemplo de la Tabla 2 anterior, el codificador de vídeo 20 puede configurarse para señalar la modalidad de CopiarDesdeArriba, seguida por datos que indican un valor de racha de seis. Esta técnica puede tener la ventaja de aumentar el tamaño de las rachas, lo que puede dar como resultado una compresión de datos mejorada.

[138] De acuerdo a aspectos de esta divulgación, durante la racha de índices codificados con relación a índices de otra fila, si una muestra de la racha tiene una muestra situada directamente encima de la muestra que está codificada como muestra de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir que la muestra está codificada como una muestra de escape. En este ejemplo, el valor real de la muestra se puede señalar a continuación de la racha.

[139] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no usar la modalidad CopiarDesdeArriba para una muestra si la muestra vecina anterior (por ejemplo, la muestra situada directamente encima de la muestra que se está codificando) está codificada como una muestra de escape. En casos en los que se permite la inclusión de muestras codificadas como muestras de escape en una racha asociada a la modalidad de paleta CopiarDesdeArriba (como se ha descrito anteriormente), se pueden eliminar algunas redundancias asociadas a la señalización de la modalidad de Escape. En un primer ejemplo, según aspectos de esta divulgación, si la muestra que sigue a una racha de índices tiene una muestra adyacente superior que está codificada como una muestra de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no codificar lo siguiente muestra como una muestra de escape. En este ejemplo, la muestra no puede codificarse como una muestra de escape, porque, de lo contrario, se incluiría en la racha de CopiarDesdeArriba. En este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para omitir la codificación de un indicador de escape o de un índice de escape, porque el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir que la muestra no está codificada como una muestra de escape.

[140] Por las mismas razones que en el ejemplo descrito anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no codificar una muestra usando la modalidad CopiarDesdeArriba después de una racha de CopiarDesdeArriba. Por consiguiente, el codificador de vídeo 20 y el decodificador 30 pueden configurarse para omitir la codificación de un elemento sintáctico relacionado con la modalidad (por ejemplo, un indicador que diferencia entre las modalidades CopiarDesdeArriba o Valor), y el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir que la muestra está codificada en modalidad de Valor.

[141] En un segundo ejemplo, si la última muestra incluida en la racha de CopiarDesdeArriba es una muestra de escape y la siguiente muestra está codificada con la modalidad de Valor, el codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar para ajustar el índice de la modalidad de Valor (así como el máximo número posible de índices que se puede usar en la binarización truncada para codificar el índice de Valor), verificando el valor del índice de la muestra vecina superior para la siguiente muestra. En este caso, si el píxel vecino superior tiene un índice, entonces el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reducir el índice de Valor en uno si el valor del índice es mayor que el índice superior. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reducir en uno el máximo número posible de índices para codificar el índice. En la etapa de reconstrucción, el decodificador de vídeo 30 puede verificar el valor del índice decodificado para determinar si el índice es mayor o igual que el índice vecino superior y, si es así, aumentar el valor decodificado en uno.

[142] La FIG. 2 es un diagrama de bloques que ilustra un codificador ejemplar de vídeo 20 que puede implementar las técnicas de esta divulgación. La FIG. 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.

[143] 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 paleta o codificación no basada en paleta. Las modalidades de codificación no basadas en paletas pueden referirse a varias modalidades de codificación temporal inter-predictiva o a modalidades de codificación espacial intra-predictiva, 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 píxel, seleccionar valores de píxel en una paleta para representar valores de píxeles de al menos algunas ubicaciones de píxel en un bloque de datos de vídeo, y señalar información que asocie al menos algunas de las ubicaciones de píxeles en el bloque de datos de vídeo a entradas en la paleta correspondientes, respectivamente, a los valores de píxel seleccionados en la paleta. La información señalizada puede ser utilizada por el decodificador de vídeo 30 para decodificar datos de vídeo.

[144] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una unidad de procesamiento de predicción 100, una memoria de datos de vídeo 101, una unidad de generación residual 102, una unidad de procesamiento de transformaciones 104, una unidad de cuantización 106, una unidad de cuantización inversa 108, una unidad de procesamiento de transformación inversa 110, una unidad de reconstrucción 112, una unidad de filtro 114, una memoria intermedia de imágenes decodificadas 116 y una unidad de codificación por entropía 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de inter-predicción 120 y una unidad de procesamiento de intra-predicción 126. La unidad de procesamiento de inter-predicción 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 basadas en paletas descritas en esta divulgación. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

5 **[145]** La memoria de datos de vídeo 101 puede almacenar datos de vídeo a ser codificados por los componentes del codificador de vídeo 20. Los datos de vídeo almacenados en la memoria de datos de vídeo 101 se pueden obtener, por ejemplo, del origen de vídeo 18. La memoria intermedia de imágenes decodificadas 116 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para su uso en la codificación de datos de vídeo mediante el codificador de vídeo 20, por ejemplo, en modalidades de intra-codificación o inter-codificación. La memoria de datos de vídeo 101 y la memoria intermedia de imágenes decodificadas 116 pueden estar formadas por cualquiera entre una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 101 y la memoria intermedia de imágenes decodificadas 116 pueden ser proporcionadas por el mismo dispositivo de memoria o dispositivos de memoria independientes. En varios ejemplos, la memoria de datos de vídeo 101 puede estar en un chip con otros componentes del codificador de vídeo 20, o fuera de chip con respecto a esos componentes.

20 **[146]** 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 división 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 las CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un CTB asociado a una CTU en cuatro sub-bloques de igual tamaño, dividir uno o más de los sub-bloques en cuatro sub-sub-bloques de igual tamaño, etc.

25 **[147]** El codificador de vídeo 20 puede codificar las CU de una CTU para generar representaciones codificadas de las CU (es decir, CU codificadas). Como parte de la codificación de una CU, la unidad de procesamiento de predicción 100 puede dividir los bloques de codificación asociados a la CU entre una o más PU de la CU. Por 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 varios 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 de predicción de luma de la PU. Suponiendo que el tamaño de una CU particular sea $2N \times 2N$, 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 intra-predicción, y a tamaños de PU simétricas de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ o similares para la inter-predicción. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden dar soporte a la división asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$, para la inter-predicción.

40 **[148]** La unidad de procesamiento de inter-predicción 120 puede generar datos predictivos para una PU realizando una inter-predicción en cada PU de una CU. Los datos predictivos para la PU pueden incluir bloques predictivos de la PU e información de movimiento para la PU. La unidad de inter-predicción 121 puede realizar diferentes operaciones para una PU de una CU según que la PU está en un fragmento I, un fragmento P o un fragmento B. En un fragmento I, todas las PU son intra-predichas. Por lo tanto, si la PU está en un fragmento I, la unidad de inter-predicción 121 no realiza inter-predicción en la PU. Por lo tanto, para bloques codificados en la modalidad I, el bloque predictivo se forma usando predicción espacial a partir de bloques contiguos previamente codificados dentro de la misma trama.

50 **[149]** Si una PU está en un fragmento P, la unidad de estimación de movimiento de la unidad de procesamiento de inter-predicción 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 contiene 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 proporciona 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 inter-predicción 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.

65 **[150]** Si una PU está en un fragmento B, la unidad de estimación de movimiento puede realizar uni-predicción o bi-predicción para la PU. Para realizar la uni-predicción 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 indica una posición en la ListalmagenRef0 o la ListalmagenRef1 de la imagen de referencia que contiene la región de referencia, un vector de movimiento que indica un desplazamiento espacial entre un bloque de predicción de la PU y una ubicación de referencia asociada a la región de referencia, y uno o más indicadores de dirección de predicción que indican si la imagen de referencia está en la ListalmagenRef0 o la ListalmagenRef1. La unidad de compensación de movimiento de la unidad de procesamiento de inter-predicción 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.

[151] Para realizar la inter-predicción 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 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 indican 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 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.

[152] De acuerdo a diversos ejemplos de esta divulgación, el codificador de vídeo 20 puede configurarse para realizar una codificación basada en paletas. Con respecto al entorno de la HEVC, como ejemplo, las técnicas de codificación basadas en paletas se pueden configurar para usarse como una modalidad de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basadas en paletas pueden configurarse para usarse como una modalidad de PU en el entorno de la HEVC. De acuerdo a esto, 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 basadas en paletas, descritas en el presente documento, ya que tales 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 paletas pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[153] La unidad de codificación basada en paleta 122, por ejemplo, puede realizar una decodificación basada en paleta cuando se selecciona una modalidad de codificación basado en paleta, por ejemplo, para una CU o PU. Por ejemplo, la unidad de codificación basada en paleta 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 paleta 122, algunas de, o todas, estas funciones pueden ser realizadas por otras unidades de procesamiento, o una combinación de diferentes unidades de procesamiento.

[154] De acuerdo a aspectos de esta divulgación, la unidad de codificación basada en paletas 122 puede configurarse para realizar cualquier combinación de las técnicas para la codificación por paletas descritas en el presente documento. Por ejemplo, como se ha descrito con respecto a las FIG. 4 a 7 a continuación, de acuerdo a aspectos de esta divulgación, la unidad de codificación basada en paletas 122 puede configurarse para reiniciar una paleta predictora para construir una paleta para un bloque de datos de vídeo basándose en uno o más bloques de otra línea. Por ejemplo, como se describe en el presente documento, la unidad de codificación basada en paletas 122 puede determinar una primera paleta para un primer bloque de datos de vídeo que está ubicado en una primera fila de bloques. La unidad de codificación basada en paletas 122 también puede generar una paleta predictora cuando codifica uno o más bloques en la primera fila. Al codificar un bloque en una segunda fila, la unidad de codificación basada en paletas 122 puede reiniciar la paleta predictora para determinar una paleta del bloque en la segunda fila, basándose en las entradas de la primera paleta.

[155] En otro ejemplo, la unidad de codificación basada en paletas 122 se puede configurar para codificar una racha de índices que incluye muestras codificadas usando más de una modalidad de paleta. Por ejemplo, la unidad de codificación basada en paletas 122 puede determinar índices de paleta de una primera fila de un bloque de datos de vídeo, donde los índices de paleta incluyen uno o más índices que están asociados a un valor de color en la paleta y a un elemento sintáctico que no está asociado a un valor de color en la paleta. La unidad de codificación basada en paletas 122 también puede codificar una racha de índices de paleta de una segunda fila del bloque de datos de vídeo con relación a los índices de paleta de la primera fila, donde la racha incluye los uno o más índices que están asociados a un valor de color en la paleta y el elemento sintáctico que no está asociado a un valor de color en la paleta.

- 5 [156] La unidad de procesamiento de intra-predicción 126 puede generar datos predictivos para una PU realizando una intra-predicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos para la PU y varios elementos sintácticos. La unidad de procesamiento de intra-predicción 126 puede realizar la intra-predicción en las PU en fragmentos I, fragmentos P y fragmentos B.
- 10 [157] Para realizar la intra-predicción en una PU, la unidad de procesamiento de intra-predicción 126 puede usar múltiples modalidades de intra-predicción para generar múltiples conjuntos de datos predictivos para la PU. La unidad de procesamiento de intra-predicción 126 puede usar muestras de bloques de muestra de las PU vecinas para generar un bloque predictivo para una PU. Las PU contiguas 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 intra-predicción 126 puede usar varios números de modalidades de intra-predicción, por ejemplo, 33 modalidades de intra-predicción direccional. En algunos ejemplos, el número de modalidades de intra-predicción puede depender del tamaño de la región asociada a la PU.
- 15 [158] La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para las PU de una CU entre los datos predictivos generados por la unidad de procesamiento de inter-predicción 120 para las PU o los datos predictivos generados por la unidad de procesamiento de intra-predicción 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 métricas de velocidad / distorsión de los conjuntos de datos predictivos. Los bloques predictivos de los datos predictivos seleccionados pueden denominarse en el presente documento bloques predictivos seleccionados.
- 20 [159] 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 de luma, Cb y Cr predictivos seleccionados de las PU de la CU, bloques residuales de luma, Cb y Cr de la CU. Por ejemplo, la unidad de generación residual 102 puede generar los bloques residuales de la CU de manera que cada muestra en los bloques residuales tenga un valor igual a una diferencia entre una muestra en un bloque de codificación de la CU y una muestra correspondiente en un correspondiente bloque predictivo seleccionado de una PU de la CU.
- 25 [160] La unidad de procesamiento de transformación 104 puede realizar la división de árbol cuádruple para dividir los bloques residuales de una CU en bloques de transformación 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 a cada una de las regiones. Las TU de una CU pueden corresponder a nodos de hoja del RQT.
- 30 [161] La unidad de procesamiento de transformación 104 puede generar bloques de coeficientes de transformación para cada TU de una CU aplicando una o más transformaciones a los bloques de transformación de la TU. La unidad de procesamiento de transformación 104 puede aplicar varias transformaciones a un bloque de transformación asociado a una TU. Por ejemplo, la unidad de procesamiento de transformación 104 puede aplicar una transformación de coseno discreta (DCT), una transformación direccional o una transformación conceptualmente similar al bloque de transformación. En algunos ejemplos, la unidad de procesamiento de transformación 104 no aplica transformaciones a un bloque de transformación. En dichos ejemplos, el bloque de transformación puede tratarse como un bloque de coeficientes de transformación.
- 35 [162] La unidad de cuantización 106 puede cuantizar los coeficientes de transformación 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 transformación de n bits puede redondearse hacia abajo hasta un coeficiente de transformación de m bits durante la cuantización, donde n es mayor que m . La unidad de cuantización 106 puede cuantizar un bloque de coeficientes asociado a una TU de una CU basándose en un valor de parámetro de cuantización (QP) asociado a la CU. El codificador de vídeo 20 puede ajustar el grado de cuantización aplicado a los bloques de coeficientes asociados a una CU, ajustando el valor del QP asociado a la CU. La cuantización puede introducir pérdida de información, por lo que los coeficientes de transformación cuantizados pueden tener una precisión inferior a los originales.
- 40 [163] La unidad de cuantización inversa 108 y la unidad de procesamiento de transformación inversa 110 pueden aplicar la cuantización inversa y transformaciones 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 transformación reconstruido asociado a una TU. Mediante la reconstrucción de bloques de transformación para cada TU de una CU de esta manera, el codificador de vídeo 20 puede reconstruir los bloques de codificación de la CU.
- 45 [164] 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 a una CU. La memoria intermedia de imágenes decodificadas
- 50
- 55
- 60
- 65

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 inter-predicción 120 puede utilizar una imagen de referencia que contiene los bloques de codificación reconstruidos para realizar la inter-predicción en las PU de otras imágenes. Además, la unidad de procesamiento de intra-predicción 126 puede usar bloques de codificación reconstruidos en la memoria intermedia de imágenes decodificadas 116 para realizar la intra-predicción en otras PU en la misma imagen que la CU.

[165] 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 adaptativa al contexto (CAVLC), una operación de CABAC, una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), una operación de codificación por entropía de división de intervalos de probabilidad (PIPE), una operación de codificación exponencial de Golomb u otro tipo de operación de codificación por entropía en los datos.

[166] Para realizar la CABAC, la unidad de codificación por entropía 118 puede seleccionar un modelo de contexto para aplicar a un cierto contexto, para codificar los símbolos a transmitir. De acuerdo a aspectos de esta divulgación, la unidad de codificación por entropía 118 puede configurarse para usar el contexto basándose en el valor de índice del valor de racha que se está codificando. Por ejemplo, la unidad de codificación por entropía 118 puede configurarse para determinar un valor de racha que indica una longitud de racha de una racha de un índice de paleta de un bloque de datos de vídeo. La unidad de codificación por entropía 118 puede configurarse para determinar un contexto para la codificación adaptativa al contexto de datos que representan el valor de racha en función del valor del índice de paleta.

[167] El codificador de vídeo 20 puede emitir un flujo de bits que incluye datos codificados por entropía, generados por la unidad de codificación por entropía 118. Por ejemplo, el flujo de bits puede incluir datos que representan un RQT para una CU.

[168] 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.

[169] 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 PU en la codificación HEVC, usando codificación basada en paletas o bien codificación no basada en paletas. Las modalidades de codificación no basadas en paletas pueden referirse a varias modalidades de codificación temporal inter-predictiva o a modalidades de codificación espacial intra-predictiva, 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 píxeles en 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 en la paleta.

[170] En el ejemplo de la FIG. 3, el decodificador de vídeo 30 incluye una unidad de decodificación por entropía 150, una memoria de datos de vídeo 151, una unidad de procesamiento de predicción 152, una unidad de cuantización inversa 154, una unidad de procesamiento de transformación inversa 156, una unidad de reconstrucción 158, una unidad de filtro 160 y una memoria intermedia de imágenes decodificadas 162. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 164 y la unidad de procesamiento de intra-predicción 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 basadas en paletas descritas en esta divulgación. En otros ejemplos, el decodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

[171] La memoria de datos de vídeo 151 puede almacenar datos de vídeo, tales como un flujo de bits de vídeo codificado, para ser decodificados por los componentes del decodificador de vídeo 30. Los datos de vídeo almacenados en la memoria de datos de vídeo 151 pueden obtenerse, por ejemplo, del medio legible por ordenador 16, por ejemplo, desde un origen de vídeo local, tal como una cámara, mediante comunicación de datos de vídeo por red cableada o inalámbrica, o accediendo a medios de almacenamiento físico de datos. La memoria de datos de vídeo 151 puede formar una memoria intermedia de imágenes codificadas (CPB) que almacena datos de vídeo

codificados a partir de un flujo de bits de vídeo codificado. La memoria intermedia de imágenes decodificadas 162 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para usar en la decodificación de datos de vídeo mediante el decodificador de vídeo 30, por ejemplo, en modalidades de intra-codificación o de inter-codificación. La memoria de datos de vídeo 151 y la memoria intermedia de imágenes decodificadas 162 pueden estar formadas por cualquiera entre varios dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 151 y la memoria intermedia de imágenes decodificadas 162 pueden proporcionarse mediante el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 151 puede estar en un chip con otros componentes del decodificador de vídeo 30, o fuera de chip con respecto a esos componentes.

[172] 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.

[173] De acuerdo a aspectos de esta divulgación, la unidad de decodificación por entropía 150 puede configurarse para usar el contexto basándose en el valor de índice del valor de racha que se está codificando. Por ejemplo, la unidad de decodificación por entropía 150 puede configurarse para determinar un valor de racha que indica una longitud de racha de una racha de un índice de paleta de un bloque de datos de vídeo. La unidad decodificadora por entropía 150 puede configurarse para determinar un contexto para la codificación adaptativa al contexto de datos que representan el valor de racha basándose en el valor del índice de paleta.

[174] La unidad de procesamiento de predicción 152, la unidad de cuantización inversa 154, la unidad de procesamiento de transformación inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 160 pueden generar datos de vídeo decodificados basándose en los elementos sintácticos extraídos del flujo de bits.

[175] Las unidades de NAL del flujo de bits pueden incluir unidades de 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 identifica un PPS asociado a una imagen que contiene el fragmento.

[176] 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.

[177] 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, decuantizar, 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 razón de compresión, es decir, la razón entre el número de bits utilizados para representar la secuencia original y la comprimida, puede controlarse ajustando el valor del QP utilizado al cuantizar los coeficientes de transformación. La razón de compresión también puede depender del procedimiento de codificación por entropía empleado.

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

[179] Si se codifica una PU usando la intra-predicción, la unidad de procesamiento de intra-predicción 166 puede realizar la intra-predicción para generar bloques predictivos para la PU. La unidad de procesamiento de intra-predicción 166 puede usar una modalidad de intra-predicción 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 intra-predicción 166 puede determinar la modalidad de intra-predicción para la PU basándose en uno o más elementos sintácticos decodificados a partir del flujo de bits.

[180] La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (ListalmagenRef0) y una segunda lista de imágenes de referencia (ListalmagenRef1) basándose en los elementos sintácticos extraídos del flujo de bits. Además, si se codifica una PU usando la inter-predicción, 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.

[181] 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 intra-predicción 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.

[182] 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 intra-predicción 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 intra-predicción o de inter-predicción en las PU de otras CU.

[183] Como se ha indicado anteriormente, el decodificador 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 selecciona 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 píxeles en 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 valores de píxel del bloque en función de los valores de píxel seleccionados en la paleta. Aunque se describen diversas funciones como realizadas por la unidad de decodificación basada en paletas 165, algunas de, o todas, estas funciones pueden ser realizadas por otras unidades de procesamiento, o por una combinación de diferentes unidades de procesamiento.

[184] La unidad de decodificación basada en paletas 165 puede recibir información de modalidad de codificación de paleta, y realizar las operaciones anteriores cuando la información de modalidad de codificación de paleta indica que la modalidad de codificación de paleta se aplica al bloque. Cuando la información de modalidad de codificación de paleta indica que la modalidad de codificación de paleta no se aplica al bloque, o cuando otra información de modalidad indica el uso de una modalidad diferente, la unidad de decodificación basada en paletas 165 decodifica el bloque de datos de vídeo utilizando una modalidad de codificación basada en paletas, por ejemplo, tal como una modalidad de codificación HEVC inter-predictiva o intra-predictiva. El bloque de datos de vídeo puede ser, por ejemplo, una CU o PU generada de acuerdo a un proceso de codificación de HEVC.

[185] De acuerdo a aspectos de esta divulgación, la unidad decodificadora basada en paletas 165 puede configurarse para realizar cualquier combinación de las técnicas para la codificación de paletas descritas en el presente documento. Por ejemplo, como se ha descrito con respecto a las FIG. 4 a 7 a continuación, de acuerdo a aspectos de esta divulgación, la unidad de decodificación basada en paletas 165 puede configurarse para reiniciar una paleta predictora para construir una paleta para un bloque de datos de vídeo basándose en uno o más bloques de otra línea. Por ejemplo, como se describe en el presente documento, la unidad de decodificación basada en paletas 165 puede determinar una primera paleta para un primer bloque de datos de vídeo que se ubica en una primera fila de bloques. La unidad de decodificación basada en paletas 165 también puede generar una paleta predictora al decodificar otros uno o más bloques en la primera fila. Al codificar un bloque en una segunda fila, la unidad de decodificación basada en paletas 165 puede reinicializar la paleta predictora para determinar una paleta del bloque en la segunda fila basándose en las entradas de la primera paleta

[186] En otro ejemplo, como se ha descrito con respecto a la FIG. 8 a continuación, la unidad de decodificación basada en paletas 165 se puede configurar para decodificar una racha de índices que incluye muestras codificadas usando más de una modalidad de paleta. Por ejemplo, la unidad decodificadora basada en paletas 165 puede determinar índices de paleta de una primera fila de un bloque de datos de vídeo, donde los índices de paleta incluyen uno o más índices que están asociados a un valor de color en la paleta y a un elemento sintáctico que no está asociado a un valor de color en la paleta. La unidad decodificadora basada en paletas 165 también puede decodificar una racha de índices de paleta de una segunda fila del bloque de datos de vídeo con relación a los índices de paleta de la primera fila, donde la racha incluye los uno o más índices que están asociados a un valor de color en la paleta y el elemento sintáctico que no está asociado a un valor de color en la paleta.

[187] 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 intra-predicción y el bloque 200 que está codificado con una modalidad de codificación de inter-predicción.

[188] Las técnicas de la FIG. 4 se describen en el contexto del codificador de vídeo 20 (FIG. 1 y FIG. 2) y el decodificador de vídeo 30 (FIG. 1 y 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.

[189] 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 de color 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.

[190] 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 Y_i , U_i y V_i . 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.

[191] 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 de 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.

[192] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar segundas paletas 192 en base a las primeras paletas 184. Por ejemplo, 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. La combinación de entradas que se utilizan para fines de predicción puede denominarse paleta predictora.

[193] En el ejemplo de la FIG. 4, cada una de las segundas paletas 192 incluye tres entradas 208 a 212 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 208 a 212 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 D, respectivamente. En este ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican qué entradas de las primeras paletas 184 (que representan una paleta predictora, aunque la paleta predictora puede incluir entradas de un determinado número de bloques) se incluyen en las segundas paletas 192.

[194] En el ejemplo de la FIG. 4, los uno o más elementos sintácticos se ilustran como un vector 216. El vector 216 tiene una serie de contenedores asociados (o bits), indicando cada contenedor si la paleta predictora asociada a ese contenedor se usa para predecir una entrada de la paleta actual. Por ejemplo, el vector 216 indica que las dos primeras entradas de las primeras paletas 184 (202 y 204) están incluidas en las segundas paletas 192 (un valor de "1" en el vector 216), mientras que la tercera entrada de las primeras paletas 184 no está incluida en las segundas paletas 192 (un valor de "0" en el vector 216). En el ejemplo de la FIG. 4, el vector es un vector Booleano. El vector se puede denominar vector de predicción de paleta.

[195] En algunos ejemplos, como se ha indicado anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar una paleta predictora (que también puede denominarse tabla de paleta predictora o lista de paleta predictora) cuando se realiza la predicción de paleta. La paleta predictora puede incluir entradas de paletas de uno o más bloques vecinos que se usan para predecir una o más entradas de una paleta para codificar un bloque actual. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden construir la lista de la misma manera. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos (tales como el vector 216) para indicar qué entradas de la paleta predictora se han de copiar a una paleta para codificar un bloque actual.

[196] De acuerdo a aspectos de esta divulgación, una paleta predictora (tal como la paleta predictora asociada al vector 216) puede reiniciarse para generar una paleta asociada a ciertos bloques en una imagen y / o tesela de datos de vídeo. Por ejemplo, como se describe con mayor detalle con respecto a los ejemplos de las FIG. 5 a 7 a continuación, una paleta predictora para construir una paleta para un bloque de datos de vídeo en una primera línea puede reiniciarse basándose en uno o más bloques de otra línea. En un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar una primera paleta para un primer bloque de datos de vídeo que se sitúa en una primera fila de bloques. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden generar una paleta predictora al codificar otros uno o más bloques en la primera fila. Al codificar un bloque en una segunda fila, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reiniciar la paleta predictora para determinar una paleta del bloque en la segunda fila basándose en las entradas de la primera paleta.

[197] La FIG. 5 es un diagrama conceptual que ilustra teselas ejemplares de acuerdo a la norma de codificación de vídeo de alta eficacia (HEVC). La HEVC contiene varias propuestas para hacer que el códec sea más compatible con el paralelismo, incluyendo las teselas y el procesamiento paralelo de frentes de onda (WPP). El borrador 10 de la HEVC define las teselas como un número entero de bloques de árbol de codificación que coexisten en una columna y una fila, ordenados consecutivamente en escaneo cuadrado de bloques de CTU de la tesela. La división de cada imagen en teselas puede denominarse una partición. Las teselas en una imagen se ordenan consecutivamente en el escaneo cuadrado de la imagen, como se muestra en la FIG. 5.

[198] Por ejemplo, la FIG. 5 ilustra un ejemplo de orden de codificación de CTB para una imagen 220 que está dividida en múltiples teselas 4, 222B, 222C, 222D, 222E, 222F, 222G, 222H y 222I (colectivamente, "teselas 4"), con límites de tesela indicados por las líneas gruesas. Cada bloque cuadrado en la imagen 220 representa un bloque de píxeles asociado a un CTB. Los números en los bloques de píxeles indican las posiciones de los CTB correspondientes (por ejemplo, las LCU) en un orden de codificación de tesela para la imagen 220.

[199] Como se ilustra en el ejemplo de la FIG. 5, los CTB en la tesela 222A se codifican primero, seguidos por los CTB en la tesela 222B, seguidos por los CTB en la tesela 222C, seguidos por los CTB en la tesela 222D, seguidos por los CTB en la tesela 222E, seguidos por los CTB en la tesela 222F, seguidos por los CTB en la tesela 222G, seguidos por los CTB en la tesela 222H, seguidos por los CTB en la tesela 222I. El orden de procesamiento de las teselas 222 se ilustra generalmente por los números incluidos en las teselas 222. Dentro de cada una de las teselas 222, los CTB se codifican de acuerdo a un orden de escaneo de cuadrado.

[200] El número de teselas y la ubicación de sus límites se pueden definir para toda la secuencia o cambiar de una imagen a otra. Los límites, de manera similar a los límites de fragmento, rompen las dependencias de análisis sintáctico y de predicción, de modo que una tesela se pueda procesar de forma independiente. Sin embargo, en algunos casos, los filtros en bucle (por ejemplo, filtros de desbloqueo y filtros de desplazamiento adaptativo de muestra (SAO)) aún pueden cruzar los límites de las teselas.

[201] El uso de teselas puede mejorar el paralelismo, ya que no se necesita ninguna comunicación (o relativamente poca) entre los procesadores o los núcleos de procesadores para la decodificación por entropía y la reconstrucción de compensación de movimiento. Además, las teselas pueden exhibir una eficacia de codificación relativamente mejor cuando se comparan con los fragmentos, porque las teselas permiten formas de partición de imágenes que contienen muestras con una correlación potencialmente superior a los fragmentos. Los mosaicos también pueden reducir la sobrecarga de la cabecera de fragmento.

[202] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reiniciar una paleta predictora para determinar una paleta de un CTB basándose en una paleta de otro CTB. En un ejemplo con fines de ilustración, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reiniciar una paleta predictora para determinar una paleta actual de un CTB actual al comienzo de cada línea de CTB. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden reiniciar la paleta predictora usando entradas de paleta de un primer CTB de la línea de CTB encima del CTB actual.

[203] En un ejemplo con fines de ilustración, después de codificar los CTB (4) (de la tesela 222A) usando codificación basada en paletas, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para generar una paleta predictora para determinar una paleta de CTB (5) (de la tesela 222A). Además, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para actualizar la paleta predictora después de codificar cada uno entre el CTB (5), la CTU (6) y la CTU (7), usando una modalidad de codificación basada en paletas. Es

decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reemplazar o agregar una o más entradas de la paleta predictora después de codificar cada uno entre el CTB (5), el CTB (6) y el CTB (7).

[204] Según aspectos de esta divulgación, al codificar el CTB (8) de la línea de CTB debajo de la línea de CTB que incluye el CTB (4) (suponiendo el orden de escaneo de cuadrículado), el codificador de vídeo 20 puede reinicializar la paleta predictora antes de generar una paleta para codificar el CTB (8). Por ejemplo, en lugar de usar la paleta predictora según lo actualizado previamente basándose en el CTB (7), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reinicializar la paleta predictora basándose en las entradas de una paleta asociada al primer CTB de la fila de CTB contigua superior, es decir, CTB (4). De esta manera, puede ser más probable que la paleta predictora para generar la paleta para el CTB (8) tenga colores similares a los utilizados en el CTB (8), porque el CTB (4) está situado espacialmente más cerca del CTB (4) que el CTB (7).

[205] Como se ha indicado anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reinicializar la paleta predictora basándose en las entradas de una paleta del primer CTB de la fila de los CTB vecinos superiores, es decir, los CTB (4). Según algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar la paleta predictora para el CTB (8) (por ejemplo, la paleta predictora utilizada para generar una paleta para codificar muestras del CTB (8), reemplazando entradas de la paleta predictora por entradas de la paleta utilizada para codificar el CTB (4).

[206] En otro ejemplo, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar la paleta predictora basándose en una paleta predictora asociada al CTB (4) que incluye entradas de la paleta del CTB (4). Por ejemplo, como se ha señalado anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden generar una paleta predictora (o actualizar una paleta predictora generada previamente) después de codificar el CTB (4). El codificador de vídeo 20 y el decodificador de vídeo 30 pueden incluir entradas del CTB (4) en la paleta predictora generada o actualizada. En este caso, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reinicializar la paleta predictora basándose en las entradas de la paleta predictora después de codificar el CTB (4) y antes de codificar el CTB (5). En este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para reinicializar la paleta predictora del CTB (8) después de terminar la codificación del CTB (4) y antes de codificar el CTB (5).

[207] Si bien el ejemplo anterior se describe con respecto al CTB (4) y al CTB (5), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aplicar técnicas generalizadas de una manera similar. Por ejemplo, con respecto al ejemplo de reinicializar la paleta predictora para el CTB (8), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para generar y / o actualizar una paleta predictora después de codificar N CTB de la línea de los CTB por encima del CTB (8) (por ejemplo, la línea contigua superior de los CTB, que incluye el CTB (4)). El codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar configurados para reinicializar la paleta predictora para el CTB (8), basándose en la paleta predictora generado (o actualizada) y antes de comenzar a codificar el $N+1$ -ésimo CTB en la línea de los CTB por encima del CTB (4). En consecuencia, en algunos ejemplos, la paleta predictora utilizada para la reinicialización puede incluir entradas que son diferentes (o además de) las entradas de la paleta para el CTB (4).

[208] En un ejemplo, el proceso descrito anteriormente para el CTB (4) se puede realizar en cambio para los CTB (5). Por ejemplo, la paleta predictora generada después del CTB (5) puede usarse en lugar del CTB (4) en el ejemplo anterior. En este caso, el CTB utilizado para el proceso de reinicialización de la paleta puede ser el mismo CTB utilizado para el proceso de inicialización del contexto de CABAC, aplicado para frentes de onda, lo que hipotéticamente podría armonizar la reinicialización de la paleta con el proceso de inicialización del contexto.

[209] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no reinicializar una paleta predictora para un CTB inicial que se encuentra en la línea superior (suponiendo un escaneo de cuadrículado de arriba a abajo) de una imagen o tesela. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no reinicializar una paleta predictora para los CTB (0) (de la tesela 222A), porque no hay ningún CTB situado por encima del CTB (0).

[210] Aunque los ejemplos de la FIG. 5 se describen con respecto a los CTB con fines de ilustración, se debería entender que las técnicas se pueden extender a cualquier tipo de bloque de datos de vídeo.

[211] La FIG. 6 es un diagrama conceptual que ilustra los frentes de onda para el procesamiento paralelo de frentes de onda (WPP). La HEVC define una técnica de WPP. Cuando el WPP está habilitado, cada fila de CTU de una imagen es una partición independiente. Sin embargo, en comparación con los fragmentos y las teselas, no se ha roto ninguna dependencia de codificación en los límites de filas de CTU. Además, las probabilidades de CABAC se propagan desde los segundos CTB de la fila anterior, para reducir aún más las pérdidas de codificación. Además, el WPP no cambia el orden regular de escaneo cuadrículado. Debido a que las dependencias no están rotas, la pérdida de velocidad-distorsión de un flujo de bits del WPP es pequeña, en comparación con un flujo de bits no paralelo.

[212] Cuando se habilita el WPP, un cierto número de procesadores, hasta el número de filas de CTU, puede funcionar en paralelo para procesar la fila (o líneas) de CTU. Sin embargo, las dependencias de frente de onda no permiten que todas las filas de CTB empiecen la decodificación al comienzo de la imagen. En consecuencia, las filas de CTB tampoco pueden terminar la decodificación al mismo tiempo al final de la imagen. Esto introduce ineficacias de paralelización que se vuelven más evidentes cuando se usa un gran número de procesadores. La FIG. 6 ilustra cómo el WPP procesa filas de los CTB en paralelo, comenzando cada fila con las probabilidades de CABAC disponibles después de procesar el segundo CTB de la fila superior.

[213] De acuerdo a aspectos de esta divulgación, como se describe en mayor detalle a continuación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar si se reinicia una paleta predictora basándose en los frentes de onda ilustrados en la FIG. 6.

[214] La FIG. 7 es un diagrama conceptual que ilustra un ejemplo de reinicio de paleta predictora en presencia de múltiples teselas y frentes de onda, congruente con las técnicas de esta divulgación. En el ejemplo de la FIG. 7, las líneas continuas relativamente más gruesas pueden indicar los límites de teselas, tales como los límites de tesela de la tesela-?30. Las líneas discontinuas indican los límites de los CTB. Por ejemplo, la fila de CTB 232 indica una fila de teselas de los CTB que incluyen los dos CTB superiores de la tesela 230.

[215] Algunas técnicas de esta divulgación incluyen reiniciar una paleta predictora basándose en un número de frentes de onda y / o un número de teselas que se usan para codificar una imagen, tal como la imagen 234. Por ejemplo, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden reiniciar una paleta predictora para los primeros CTB de tesela (tales como el CTB superior izquierdo de la tesela 230) cuando se usan múltiples frentes de onda y múltiples teselas para codificar la imagen 234 (tal como las teselas y los frentes de onda mostrados en las FIGs. 5 y 6, respectivamente). En el ejemplo de la FIG. 7, los círculos continuos incluidos dentro de los CTB indican un primer CTB de tesela de las teselas para las que se realiza el reinicio de paleta. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden reiniciar una paleta al fijar en cero el tamaño de la paleta predictora y no predecir ninguna entrada de paleta de una paleta que se está usando para codificar un bloque actual.

[216] La FIG. 8 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 FIG. 5 incluye un mapa 240 de índices que relacionan posiciones respectivas de píxeles asociados a los valores de índice para una entrada de paletas 244. Por ejemplo, el índice 1 está asociado al Valor A, el índice 2 está asociado al Valor B y el índice 3 está asociado al Valor C. Además, cuando se indican muestras de escape usando señalización de escape implícita, el codificador de vídeo 20 y el decodificador de vídeo 30 también pueden añadir un índice adicional a las paletas 244, ilustradas en la FIG. 5 como el índice 4, que puede indicar que las muestras del mapa 240 asociadas al índice 4 son muestras de escape. En este caso, 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 el mapa 240 si el valor de píxel no está incluido en las paletas 244.

[217] 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 del mapa (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.

[218] El codificador de vídeo 20 puede, en dicho ejemplo, codificar también un índice de paleta (mostrado en el ejemplo de la FIG. 8 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.

[219] 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). En consecuencia, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican una serie de valores de índice consecutivos, en un orden de escaneo dado, que tienen el mismo índice. Como se ha indicado anteriormente, la codificación de índices de esta manera se pueden denominar modalidad de paleta de valores.

[220] Como se ha indicado anteriormente, las rachas se pueden usar junto con una modalidad de CopiarDesdeArriba o de Valor. 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", dos valores de índice de "3" y una muestra que no está incluida en las paletas 244 (representada por el índice 4, aunque se puede usar un indicador de escape de nivel de muestra para la señalización de escape explícita), que se puede denominar muestra de escape.

[221] En este ejemplo, el codificador de vídeo 20 puede usar la modalidad de CopiarDesdeArriba para codificar 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.

[222] Después de codificar los uno o más elementos sintácticos que indican la primera posición de la fila 264 y la racha de dos entradas (señaladas anteriormente), el codificador de vídeo 20 puede codificar las posiciones cuarta y quinta en la fila 268 (de izquierda a derecha), usando la modalidad de Valor. Por ejemplo, el codificador de vídeo 20 puede codificar 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 ejemplo, modalidad de Valor). Por lo tanto, el codificador de vídeo 20 codifica estas dos posiciones sin referencia a otra línea.

[223] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar por contexto un valor de racha para un índice codificado usando la modalidad de Valor, basándose en el valor del índice. Por ejemplo, como se describe en el presente documento, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar una modalidad adaptativa al contexto de un proceso de codificación CABAC para codificar datos que indican un valor de racha. De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para determinar un contexto para codificar los datos basándose en el índice de la racha.

[224] En el ejemplo de la FIG. 8, el codificador de vídeo 20 y el decodificador de vídeo 30 codifican un índice de 1 para la cuarta posición señalada anteriormente del mapa 40. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden binarizar un valor de racha de 1 para la cuarta posición del mapa 240. En un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden generar un indicador que indica que el valor de racha es mayor que cero pero no mayor que uno (lo que puede indicarse usando un indicador independiente).

[225] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden seleccionar un contexto para codificar el indicador mayor que cero basándose en el valor de índice de 1. Por ejemplo, según algunos aspectos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden definir un conjunto de contextos para codificar el valor de racha. El conjunto de contextos puede incluir un contexto, dos contextos, tres contextos, cuatro contextos o más contextos.

[226] En un ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (0) basándose en que el índice sea igual a cero. En este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (1) basándose en que el índice tiene cualquier valor distinto de cero. Con este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 seleccionan un contexto de racha de (1) basándose en que el índice sea 1 para la cuarta posición del mapa 240.

[227] En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (0) basándose en que el índice sea igual a cero. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (1) basándose en que el índice sea uno. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (2) basándose en que el índice tenga cualquier valor mayor que uno. Con este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 seleccionan un contexto de racha de (1) basándose en que el índice sea 1 para la cuarta posición del mapa 240.

[228] En otro ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (0) basándose en que el índice sea igual a cero. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (1) basándose en que el índice sea uno o dos. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para seleccionar el contexto de racha (2) basándose en que el índice sea mayor que dos. Con este ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 seleccionan un contexto de racha de (1) basándose en que el índice sea 1 para la cuarta posición del mapa 240.

[229] De acuerdo a aspectos de esta divulgación, el contexto puede basarse en un valor de índice real o un valor de índice ajustado. Por ejemplo, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para determinar el contexto para codificar un valor de racha basándose en un índice de paleta señalado en un flujo de bits (que puede estar asociado al elemento sintáctico índice_paleta_ajustado), en lugar del índice de paleta realmente usado para codificar una muestra en un bloque, en casos en los que se señala un índice de paleta ajustado en el flujo de bits.

[230] Después de codificar la racha asociada a la cuarta posición del mapa 240, el codificador de vídeo 20 puede codificar entonces la primera posición que tiene un valor de índice de 3 en la fila 268 usando la modalidad de CopiarDesdeArriba con respecto a la fila superior 264. Por ejemplo, el codificador de vídeo 20 puede señalar una modalidad de CopiarDesdeArriba y una racha de 1. En consecuencia, el codificador de vídeo 20 puede seleccionar entre codificar valores de píxel o de índice de una línea con respecto a otros valores de la línea, por ejemplo, utilizando una racha, codificar valores de píxel o de índice de una línea con respecto a valores de otra línea (o columna), o una combinación de las mismas. El codificador de vídeo 20 puede, en algunos ejemplos, efectuar una optimización de velocidad / distorsión para realizar la selección.

[231] El codificador de vídeo 20 puede entonces codificar la muestra de escape para la muestra final de la fila 268 (de izquierda a derecha), que no está incluida en las primeras paletas 244. Por ejemplo, el codificador de vídeo 20 puede codificar la posición final de la fila 268 como una muestra de escape. Es decir, el codificador de vídeo 20 puede codificar una indicación de que la posición final de la fila 268 es una muestra de escape (por ejemplo, índice 4), así como una indicación del valor de muestra. El decodificador de vídeo 30 puede obtener la sintaxis descrita anteriormente desde un flujo de bits codificado y reconstruir la fila 268 usando tal sintaxis.

[232] Como se ha señalado anteriormente, puede haber dos o más técnicas para codificar una indicación de una muestra codificada como una muestra de escape. Por ejemplo, con señalización de escape explícita, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar un indicador de escape explícito para cada posición de muestra del mapa 240. Si una muestra particular (tal como la muestra final de la fila 268) se codifica como una muestra de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos que indican el valor de color para la muestra particular. Si la muestra no está codificada como una muestra de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos adicionales para indicar si la modalidad es CopiarDesdeArriba o Valor, tal como un indicador SPoint.

[233] Con la señalización de escape implícita, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden agregar un índice adicional a las paletas 244 (índice de entrada 4). El codificador de vídeo 20 y el decodificador de vídeo 30 pueden usar el índice adicional para las paletas 244 para indicar que una muestra está codificada como una muestra de escape. El índice adicional, sin embargo, no tiene un valor de color asociado. Más bien, el codificador de vídeo 20 y el decodificador de vídeo 30 también codifican valores de color para cada muestra que esté asociada al índice adicional. Si la muestra no está codificada como una muestra de escape, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos para indicar si la modalidad es CopiarDesdeArriba o Valor, tal como un indicador SPoint.

[234] De acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar una racha de índices usando la modalidad CopiarDesdeArriba que incluye una o más muestras codificadas como muestras de escape. En algunos casos, las muestras codificadas como muestras de escape pueden denominarse "índices" incluidos en la racha, a pesar de que las muestras codificadas como muestras de escape no tienen un índice que se correlacione con un valor de color de una paleta (por ejemplo, el índice adicional es agregado a la paleta con fines de indicar muestras codificadas como muestras de escape, pero no tiene un valor de color correspondiente). Por ejemplo, el codificador de vídeo 20 y el decodificador 30 pueden configurarse para copiar indicadores de escape (en la señalización de escape explícita) o un índice de escape (en la señalización de escape implícita) dentro de una racha de otros índices que tienen valores de color asociados en una paleta que se está copiando con la modalidad de paleta CopiarDesdeArriba. Tanto los indicadores de escape como el índice de escape pueden denominarse un elemento sintáctico que no está asociado a un valor de color en la paleta, porque ni el indicador ni el índice de escape se incluyen en una paleta.

[235] Por lo tanto, de acuerdo a aspectos de esta divulgación, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para deducir que una muestra actual incluida en una racha asociada a la modalidad de paleta CopiarDesdeArriba (por ejemplo, una muestra codificada en relación con una muestra de otra fila) se codifica como una muestra de escape si la muestra adyacente superior a la muestra actual se codifica como una muestra de escape. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden codificar los valores de escape relacionados para todas las muestras codificadas como muestra de escape a continuación de la racha.

[236] Con respecto a la FIG. 8, la tercera muestra de la fila 270 se codifica como una muestra de escape y la tercera muestra de la fila 272 se codifica como una muestra de escape (suponiendo un escaneo cuadrulado de izquierda a derecha). En el ejemplo de la FIG. 8, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar una primera muestra de la fila 270 usando la modalidad CopiarDesdeArriba. Además, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar un valor de racha que indica una longitud de racha de 15

para la modalidad CopiarDesdeArriba. La racha incluye tanto la tercera muestra de la fila 270 como la tercera muestra de la fila 272. En consecuencia, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar datos que indican un valor de escape (por ejemplo, un valor de color) para la tercera muestra de la fila 270. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden codificar datos que indican un valor de escape (por ejemplo, un valor de color) para la tercera muestra de la fila 272, que puede ser diferente al valor de escape de la fila 270.

[237] La FIG. 9 es un diagrama de flujo que ilustra un proceso ejemplar para la determinación de una paleta predictora, congruente con las técnicas de esta divulgación. El proceso de la FIG. 9 generalmente se describe como realizado por un codificador de vídeo, que puede incluir un codificador de vídeo 20, un decodificador de vídeo 30 o varios otros procesadores.

[238] En el ejemplo de la FIG. 9, el codificador de vídeo puede determinar una primera paleta para un primer bloque de datos de vídeo que se ubica en una primera fila de bloques (290). Si bien se describe con respecto a "filas" con fines de ilustración, debería entenderse que las técnicas también pueden aplicarse a columnas de bloques en casos en que el codificador de vídeo escanea bloques en orden de columnas. En cualquier caso, la primera paleta para el primer bloque puede incluir una o más entradas de paleta que tienen, cada una, un índice de paleta que está asociado a un valor de color para codificar el bloque.

[239] El codificador de vídeo también genera una paleta predictora para generar o construir al menos una paleta de al menos un segundo bloque que se sitúa en la primera fila (292). Por ejemplo, el codificador de vídeo puede generar una paleta predictora basándose en las entradas de la primera paleta después de codificar el primer bloque. El codificador de vídeo también puede actualizar la paleta predictora para cada bloque codificado posteriormente de la primera fila, que se codifique utilizando codificación basada en paletas.

[240] De acuerdo a aspectos de esta divulgación, el codificador de vídeo puede reinicializar la paleta predictora para determinar una tercera paleta de un tercer bloque que se sitúa en una segunda fila de bloques (294). El tercer bloque puede ser el primer bloque que se codifica en la segunda fila de bloques. El tercer bloque puede colocarse inmediatamente debajo del primer bloque.

[241] El codificador de vídeo puede reiniciar la paleta predictora basándose en las entradas de la paleta del primer bloque. Según algunos ejemplos, el codificador de vídeo puede configurarse para reinicializar la paleta predictora reemplazando las entradas de la paleta predictora por las entradas de la primera paleta. En algunos ejemplos, el codificador de vídeo puede configurarse para reinicializar la paleta predictora usando la paleta predictora que se genera después de codificar el primer bloque (por ejemplo, antes de codificar un segundo bloque en la primera fila) o una paleta predictora actualizada, tal como actualizada después de la codificación de uno o más bloques de la primera fila.

[242] El codificador de vídeo puede determinar la tercera paleta basándose en la paleta predictora reinicializada (296). Por ejemplo, el codificador de vídeo puede determinar qué entradas de la paleta predictora copiar a la tercera paleta y copiar las entradas relevantes. En algunos ejemplos, como se describe en este documento, el codificador de vídeo puede determinar las entradas de la paleta predictora para copiar basándose en un vector de predicción.

[243] El codificador de vídeo puede entonces codificar / decodificar el bloque usando la tercera paleta (298). Por ejemplo, en casos en los que el codificador de vídeo funciona como un codificador de vídeo (tal como el codificador de vídeo 20), el codificador de vídeo puede codificar datos que representan los índices del tercer bloque usando la paleta determinada en un flujo de bits codificado. En casos en los que el codificador de vídeo funciona como un decodificador de vídeo (tal como el decodificador de vídeo 30), el decodificador de vídeo puede determinar valores de muestra para los índices del tercer bloque usando la tercera paleta.

[244] La FIG. 10 es un diagrama de flujo que ilustra un proceso ejemplar para codificar un valor de racha de una racha de índices de paleta, congruente con las técnicas de esta divulgación. El proceso de la FIG. 10 generalmente se describe como realizado por un codificador de vídeo, que puede incluir un codificador de vídeo 20, un decodificador de vídeo 30 o varios otros procesadores.

[245] En el ejemplo de la FIG. 10, el codificador de vídeo determina un valor de racha que indica una longitud de racha de una racha de un índice de paleta (310). Por ejemplo, el codificador de vídeo puede codificar una indicación de un índice de paleta, por ejemplo, usando una modalidad Valor de paleta. El codificador de vídeo también puede determinar el número de muestras consecutivas, en un orden de escaneo, que tienen el mismo valor que el índice codificado. La cantidad de muestras consecutivas puede ser el valor de la racha.

[246] El codificador de vídeo también determina el contexto para la codificación adaptativa al contexto de los datos determinados que indican el valor de racha basándose en el índice de paleta (312). Por ejemplo, el codificador de vídeo puede determinar uno o más contextos para la codificación CABAC del valor de racha determinado basándose en el índice de paleta. En algunos ejemplos, el codificador de vídeo puede seleccionar entre una pluralidad de contextos definidos para codificar el valor de la racha.

[247] El codificador de vídeo puede entonces codificar / decodificar los datos que indican el valor de racha usando el contexto determinado (314). Por ejemplo, en casos en los que el codificador de vídeo funciona como un codificador de vídeo (tal como el codificador de vídeo 20), el codificador de vídeo puede codificar los datos que indican el valor de racha en un flujo de bits codificado. En casos en los que el codificador de vídeo funciona como un decodificador de vídeo (tal como un decodificador de vídeo 30), el decodificador de vídeo puede analizar sintácticamente (decodificar) los datos que indican el valor de racha a partir de un flujo de bits codificado.

[248] La FIG. 11 es un diagrama de flujo que ilustra un proceso ejemplar para codificar una racha de índices codificados con más de una modalidad de codificación de paleta, congruente con las técnicas de esta divulgación. El proceso de la FIG. 11 generalmente se describe como realizado por un codificador de vídeo, que puede incluir un codificador de vídeo 20, un decodificador de vídeo 30 u otros varios procesadores.

[249] El codificador de vídeo determina los índices de una primera fila de un bloque que incluye índices asociados a un color de una paleta y uno o más elementos sintácticos no asociados a un color de la paleta (320). Por ejemplo, el codificador de vídeo puede determinar muestras para las cuales los índices de paleta están codificados y muestras codificadas en la modalidad de paleta Escape.

[250] El codificador de vídeo también puede codificar / decodificar una racha de índices de una segunda fila con relación a los índices de paleta de la primera fila (322). Por ejemplo, en casos en los que el codificador de vídeo funciona como un codificador de vídeo (tal como un codificador de vídeo 20), el codificador de vídeo puede codificar una indicación de la modalidad CopiarDesdeArriba y una indicación de la longitud de racha de la racha. En casos en los que el codificador de vídeo funciona como un decodificador de vídeo (tal como un decodificador de vídeo 30), el decodificador de vídeo puede decodificar una indicación de la modalidad CopiarDesdeArriba y una indicación de la longitud de racha de la racha. En cualquier caso, la ejecución incluye tanto los índices que tienen colores asociados como los uno o más elementos sintácticos no asociados a un color de la paleta, por ejemplo, muestras codificadas en la modalidad CopiarDesdeArriba y como muestras de escape.

[251] El codificador de vídeo puede entonces codificar / decodificar datos que indican valores de color para elementos sintácticos no asociados a un color de la paleta (324). Por ejemplo, en casos en que el codificador de vídeo funciona como un codificador de vídeo (tal como un codificador de vídeo 20), el codificador de vídeo puede codificar datos que indican los valores de color de las muestras representadas por los elementos sintácticos en un flujo de bits codificado. En casos en los que el codificador de vídeo funciona como un decodificador de vídeo (tal como el decodificador de vídeo 30), el decodificador de vídeo puede decodificar datos que indican los valores de color de las muestras representadas por los elementos sintácticos de un flujo de bits.

[252] Debería entenderse que todas las técnicas descritas en este documento pueden usarse individualmente o en combinación. Esta divulgación incluye varios procedimientos de señalización que pueden cambiar en función de ciertos factores tales como el tamaño de bloque, el tamaño de paleta, el tipo de fragmento, etc. Tal variación en la señalización o la deducción de los elementos sintácticos puede ser conocida por el codificador y decodificador a priori o puede ser señalizada explícitamente en el conjunto de parámetros de vídeo (VPS), el conjunto de parámetros de secuencia (SPS), el conjunto de parámetros de imagen (PPS), la cabecera de fragmento, en un nivel de tesela o en otro lugar.

[253] Ha de reconocerse que, según el ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, fundirse 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 ciertos ejemplos, los actos o sucesos pueden realizarse simultáneamente, por ejemplo, mediante el procesamiento de múltiples hebras, el procesamiento de interrupciones o múltiples procesadores, en lugar de secuencialmente. Además, aunque ciertos aspectos de esta divulgación se describen como realizados por un único módulo o unidad, con fines de claridad, se debería entender que las técnicas de esta divulgación pueden ser realizadas por una combinación de unidades o módulos asociados a un codificador de vídeo.

[254] 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.

[255] 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.

[256] Aunque se han descrito 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

ejemplares y pueden abarcar cualquier combinación concebible de los diversos aspectos de las técnicas descritas en esta divulgación.

5 **[257]** En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse, como una o más instrucciones o código, en un medio legible por ordenador o transmitirse a través de este, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible, tal como medios de almacenamiento de datos o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo a un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) medios de almacenamiento tangibles legibles por ordenador que son no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser medios disponibles cualesquiera a los que se puede 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.

20 **[258]** A modo de ejemplo, y no de limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento 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 debidamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde una sede de la Red, un servidor u otro origen remoto 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, tal como se utiliza 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 habitualmente reproducen datos magnéticamente, mientras que otros discos reproducen datos ópticamente con láseres. Las combinaciones de los anteriores deberían incluirse también dentro del alcance de los medios legibles por ordenador.

35 **[259]** 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 lógicas programables sobre el terreno (FPGA) u otros circuitos lógicos equivalentes, integrados o discretos. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse 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 puede proporcionarse dentro de hardware dedicado y/o módulos de software, configurados para la codificación y la decodificación, o incorporarse en un códec combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

45 **[260]** Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, que incluyen un equipo manual inalámbrico, un circuito integrado (IC) o un conjunto de IC (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 pueden combinarse en una unidad de hardware de códec o proporcionarse por medio de un grupo de unidades de hardware interoperativas, que incluyen uno o más procesadores como los descritos anteriormente, conjuntamente con software y/o firmware adecuados.

55 **[261]** Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento de procesamiento de datos de vídeo, comprendiendo el procedimiento:
 - 5 determinar una primera paleta para un primer bloque de datos de vídeo que es el bloque inicial situado en una primera fila de bloques, en donde la primera paleta comprende una o más entradas de paleta, incluyendo cada una un índice de paleta que está asociado a un valor de color para codificar el bloque de datos de vídeo;
 - 10 generar una paleta predictora para construir una pluralidad de segundas paletas de una pluralidad de segundos bloques de datos de vídeo en la primera fila de bloques codificados después del primer bloque, en donde la paleta predictora incluye al menos una entrada de paleta proveniente de uno o más bloques de la primera fila que no sean el primer bloque, estando dichos uno o más bloques codificados entre el primer bloque y el segundo bloque para los que se construye la paleta predictora;
 - 15 reinicializar, basándose en un tercer bloque de datos de vídeo, situado en una segunda fila de bloques, la paleta predictora para determinar una tercera paleta del tercer bloque de datos de vídeo que se ubica en la segunda fila de bloques, en donde reinicializar la paleta predictora comprende eliminar todas las entradas de la paleta predictora e incluir al menos una de las una o más entradas de paleta de la primera paleta, o una paleta predictora inicial generada después de codificar el primer bloque, en donde la primera fila de bloques se coloca encima de la segunda fila de bloques en una imagen, y en donde el primer bloque está situado por encima del tercer bloque y un bloque a la derecha del tercer bloque;
 - 20 determinar la tercera paleta del tercer bloque basándose en la paleta predictora reinicializada; y
 - 25 codificar el tercer bloque usando la tercera paleta, en donde la codificación comprende codificación o decodificación.
2. El procedimiento de la reivindicación 1, que comprende además:
 - 30 antes de codificar el tercer bloque, codificar el primer bloque y la pluralidad de segundos bloques; y
 - 35 generar la paleta predictora inicial después de codificar el primer bloque y un bloque inicial de la pluralidad de segundos bloques que sigue inmediatamente al primer bloque en orden de codificación, y antes de codificar cualquier otro bloque de la pluralidad de segundos bloques.
3. El procedimiento de la reivindicación 1, en el que reinicializar la paleta predictora, basándose en la una o más entradas de paleta de la primera paleta o la paleta predictora inicial, comprende copiar las una o más entradas de la primera paleta, o las entradas de la paleta predictora inicial, a la paleta predictora.
4. El procedimiento de la reivindicación 1, en el que el primer bloque comprende un primer bloque de árbol codificado (CTB) y el tercer bloque comprende un tercer CTB.
5. El procedimiento de la reivindicación 1, en el que la primera fila de bloques comprende una fila superior de bloques de una tesela, comprendiendo además el procedimiento:
 - 45 determinar que una paleta predictora para predecir la primera paleta tiene cero entradas; y
 - 50 determinar la primera paleta sin predecir las una o más entradas de la primera paleta.
6. El procedimiento de la reivindicación 1, que comprende además:
 - 55 después de codificar el tercer bloque, determinar un número de frentes de onda para codificar una pluralidad de terceras filas de bloques, en donde la pluralidad de terceras filas de bloques se divide en una pluralidad de teselas;
 - basándose en que el número de frentes de onda sea más de uno, reiniciar en cero una paleta predictora asociada a cada bloque inicial de cada una entre la pluralidad de filas.
7. El procedimiento e la reivindicación 6, que comprende además:
 - 60 basándose en que el número de frentes de onda sea uno o cero, no reiniciar en cero la paleta predictora asociada a cada bloque inicial de cada una entre la pluralidad de filas.
8. El procedimiento de la reivindicación 1, en el que la codificación comprende codificación, y en el que la codificación del tercer bloque usando la tercera paleta comprende:
 - 65

determinar los valores de índice respectivos para una o más muestras del tercer bloque, en donde los respectivos valores de índice identifican una entrada de la tercera paleta; y

5 codificar los valores de índice en un flujo de bits codificado.

9. El procedimiento de la reivindicación 1, en el que la codificación comprende la decodificación, y en el que la decodificación del tercer bloque usando la tercera paleta comprende:

10 obtener, a partir de un flujo de bits codificado, respectivos valores de índice para una o más muestras del tercer bloque, en donde los respectivos valores de índice identifican una entrada de la tercera paleta; y

determinar valores para las una o más muestras apareando uno o más de los respectivos valores de índice con al menos una entrada de la paleta.

15 10. Un aparato para procesar datos de vídeo, comprendiendo el aparato:

20 medios para determinar una primera paleta para un primer bloque de datos de vídeo que es el bloque inicial ubicado en una primera fila de bloques, en donde la primera paleta comprende una o más entradas de paleta, incluyendo cada una un índice de paleta que está asociado a un valor de color para codificar el bloque de datos de vídeo;

25 medios para generar una paleta predictora para construir una pluralidad de segundas paletas de una pluralidad de segundos bloques de datos de vídeo en la primera fila de bloques codificados después del primer bloque, en donde la paleta predictora incluye al menos una entrada de paleta proveniente de uno o más bloques de la primera fila que no sean el primer bloque, estando dichos uno o más bloques codificados entre el primer bloque y el segundo bloque para los que se construye la paleta predictora;

30 medios para reinicializar, basándose en un tercer bloque de datos de vídeo que se encuentra en una segunda fila de bloques, la paleta predictora para determinar una tercera paleta del tercer bloque de datos de vídeo que se ubica en la segunda fila de bloques, en donde reinicializar la paleta predictora comprende eliminar todas las entradas de la paleta predictora e incluir al menos una de las una o más entradas de paleta de la primera paleta, o una paleta predictora inicial generada después de codificar el primer bloque, en donde la primera fila de bloques se coloca encima de la segunda fila de bloques en una imagen, y en donde el primer bloque está situado por encima del tercer bloque y un bloque a la derecha del tercer bloque;

35 medios para determinar la tercera paleta del tercer bloque basándose en la paleta predictora reinicializada; y

40 medios para codificar el tercer bloque usando la tercera paleta.

11. El aparato de la reivindicación 10, que comprende además:

45 medios para codificar, antes de codificar el tercer bloque, el primer bloque y la pluralidad de segundos bloques;

50 medios para generar la paleta predictora inicial después de codificar el primer bloque y un bloque entre la pluralidad de segundos bloques que sigue inmediatamente al primer bloque en orden de codificación, y antes de codificar cualquier otro bloque entre la pluralidad de segundos bloques.

12. El aparato de la reivindicación 10, en el que los medios para codificar comprenden un codificador.

55 13. El aparato de la reivindicación 10, en el que el medio para la codificación comprende un decodificador.

14. Un medio no transitorio legible por ordenador que tenga instrucciones almacenadas en el mismo que, al ejecutarse, hacen que uno o más procesadores realicen el procedimiento según una cualquiera de las reivindicaciones 1 a 9.

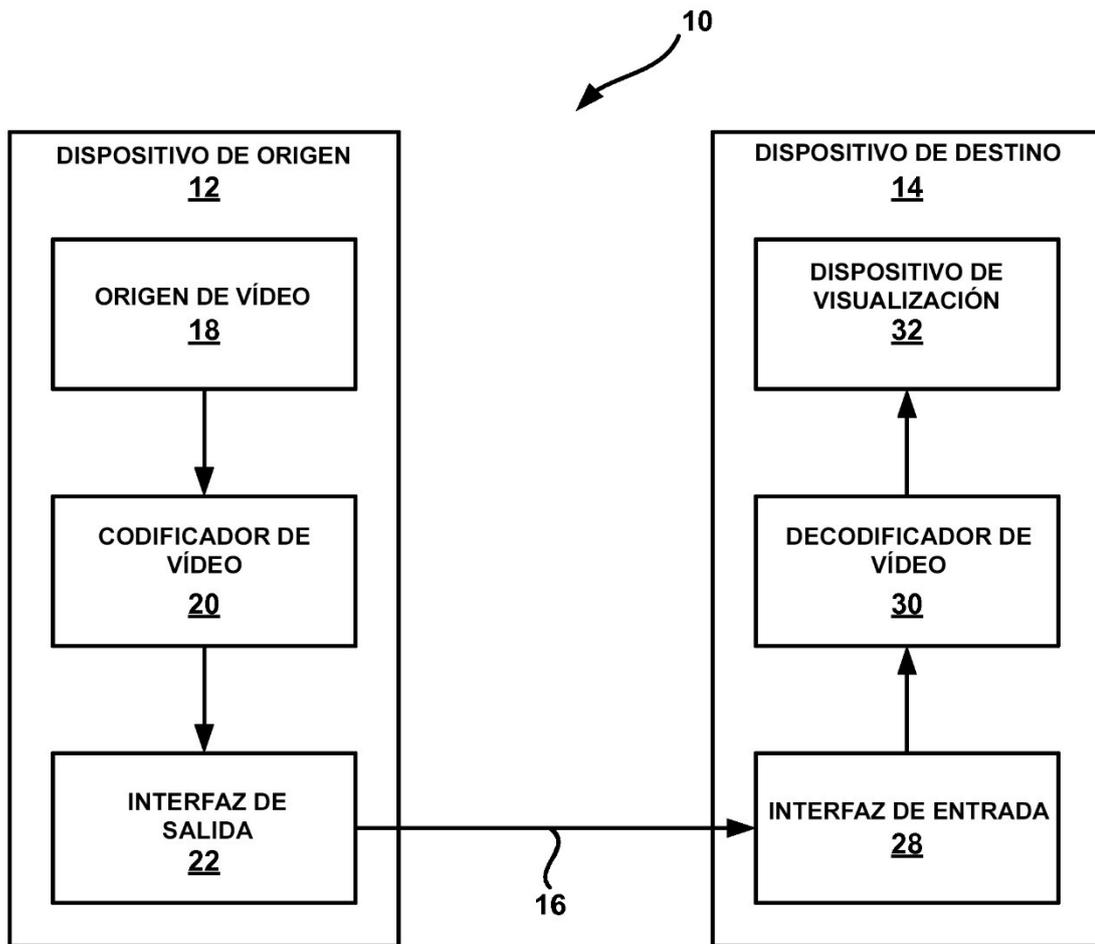


FIG. 1

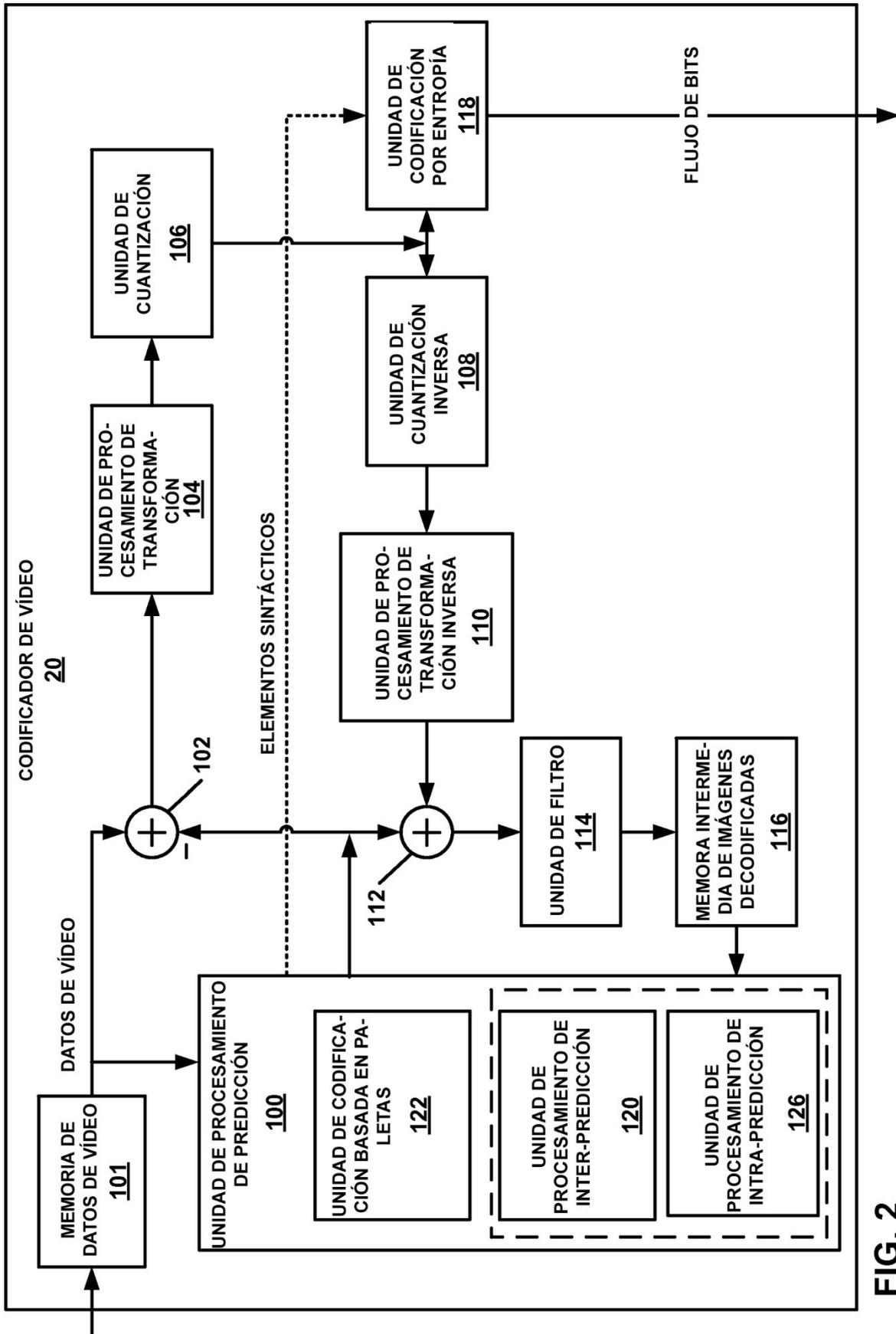


FIG. 2

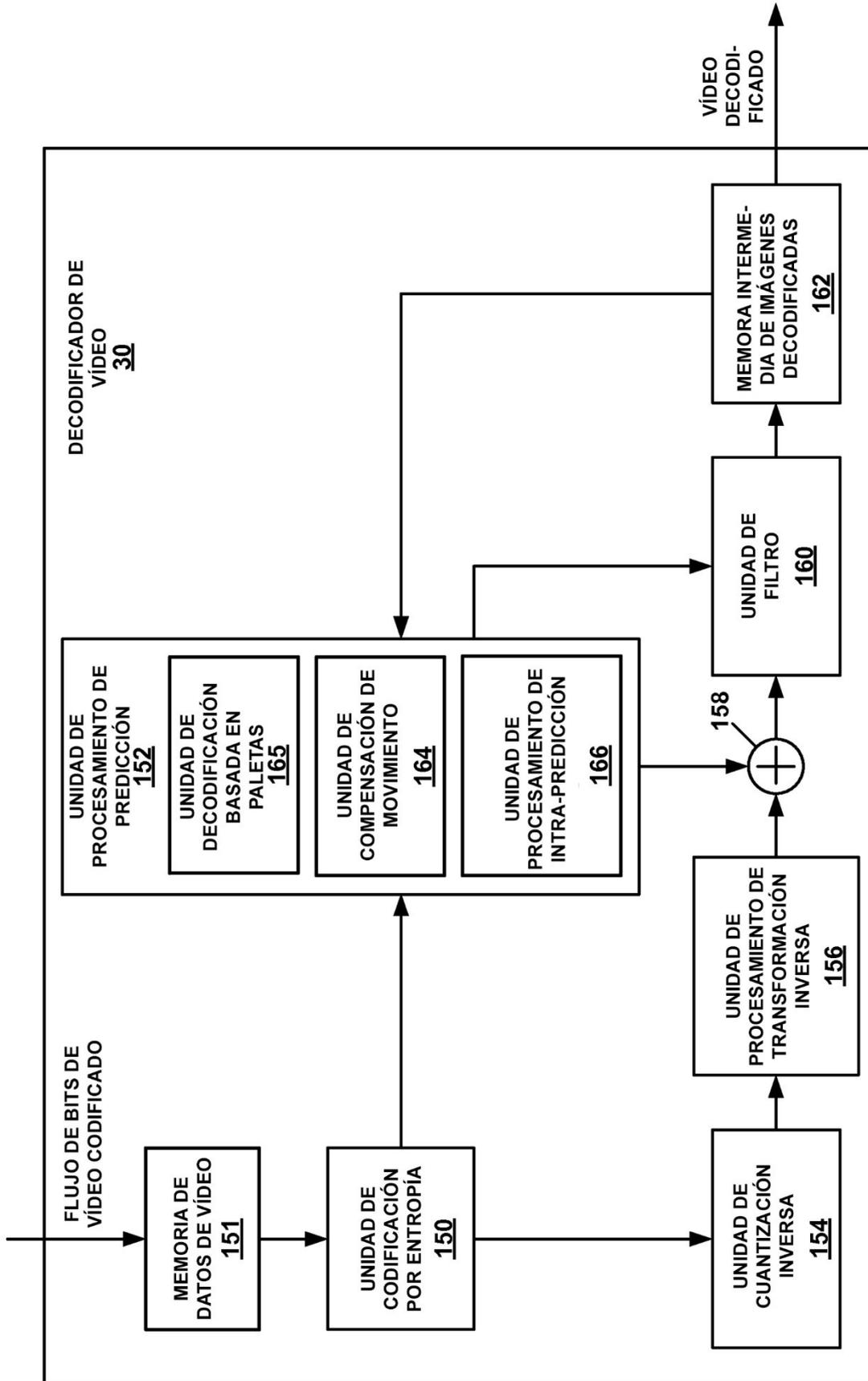


FIG. 3

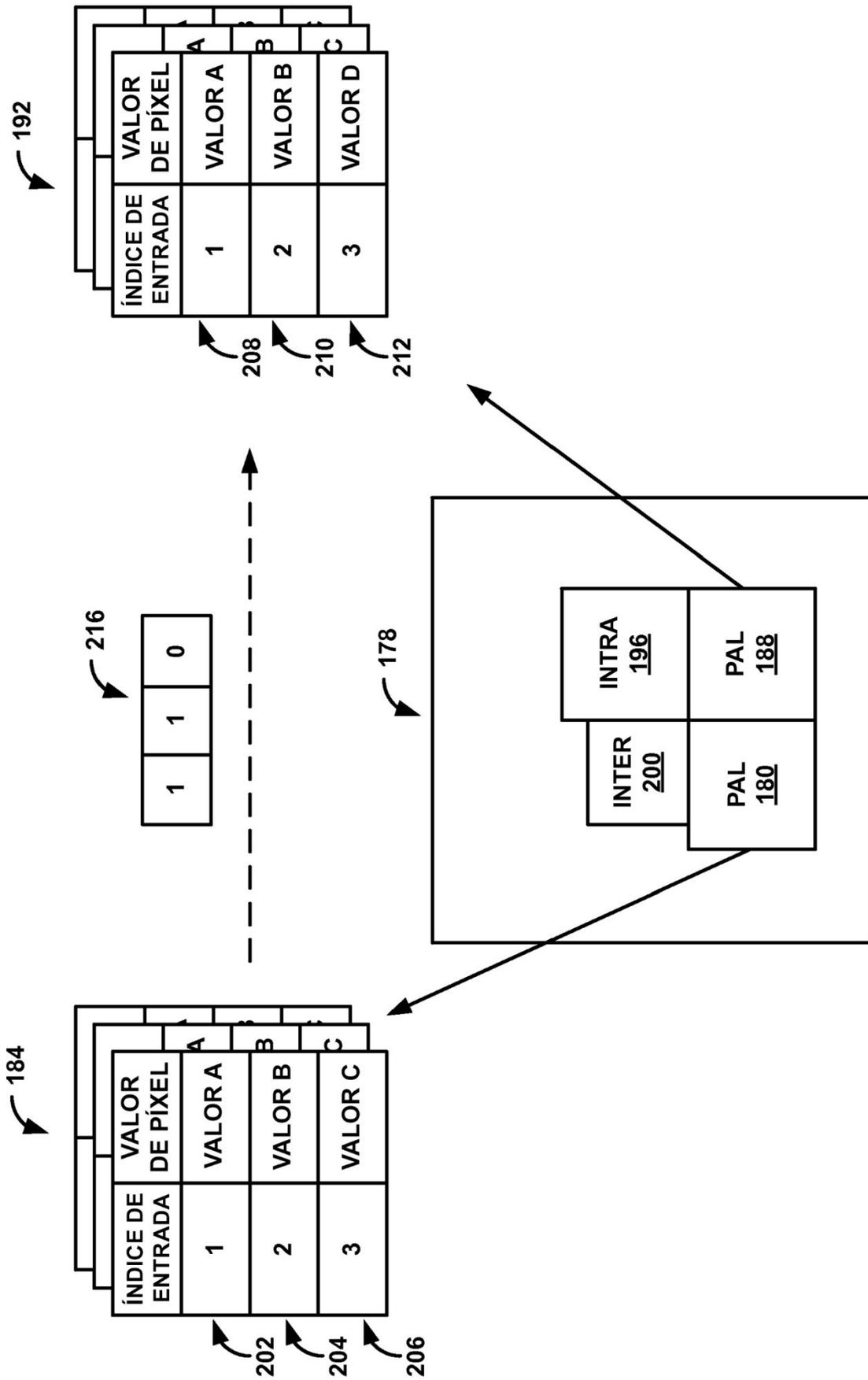


FIG. 4

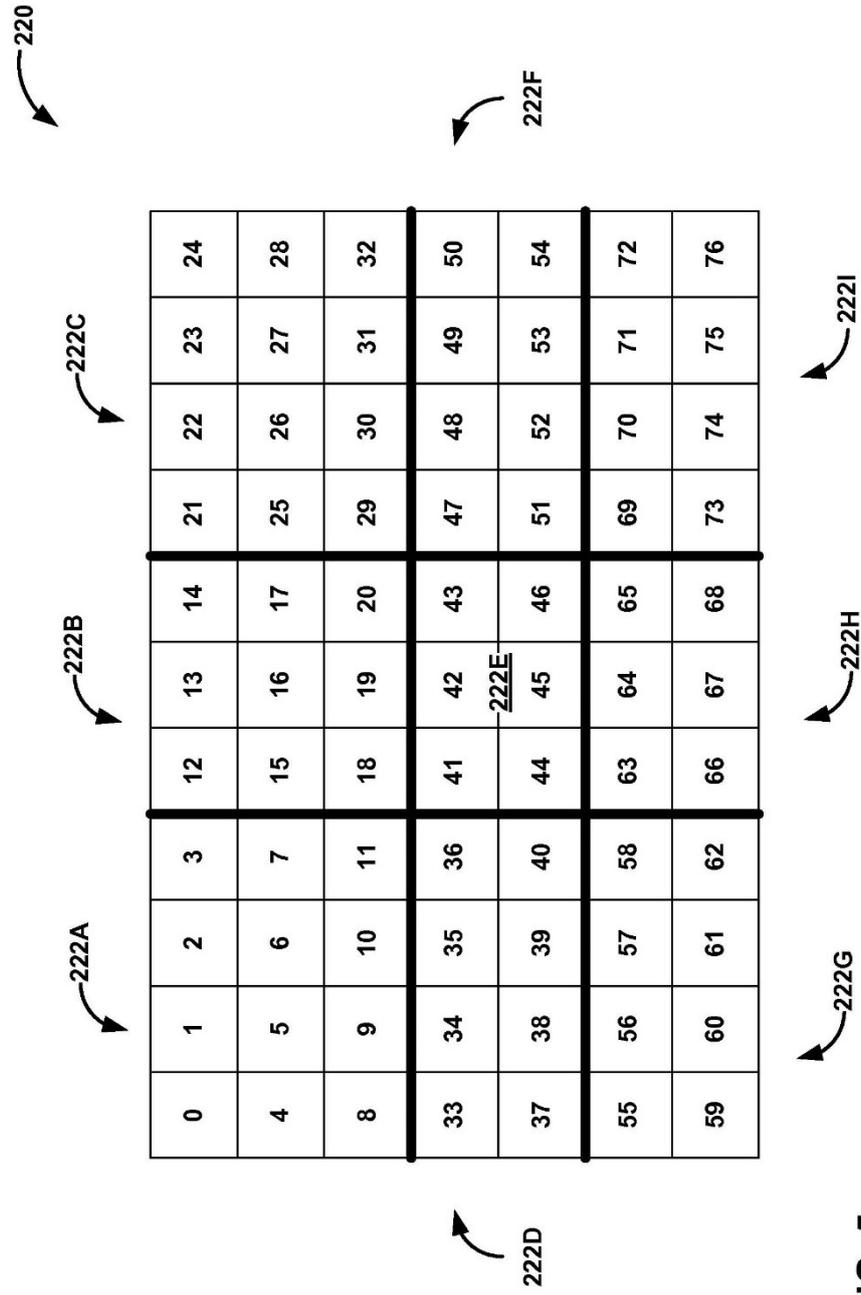


FIG. 5

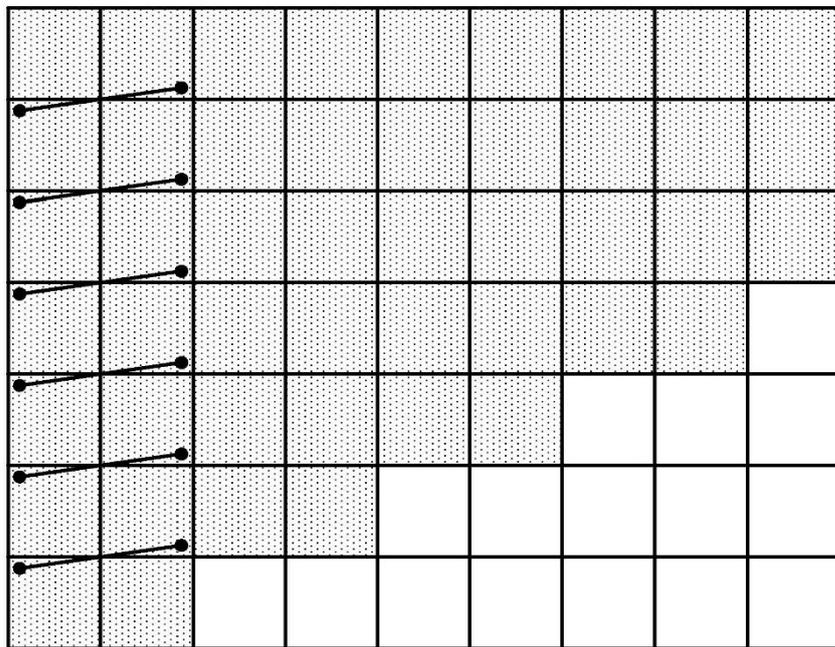


FIG. 6

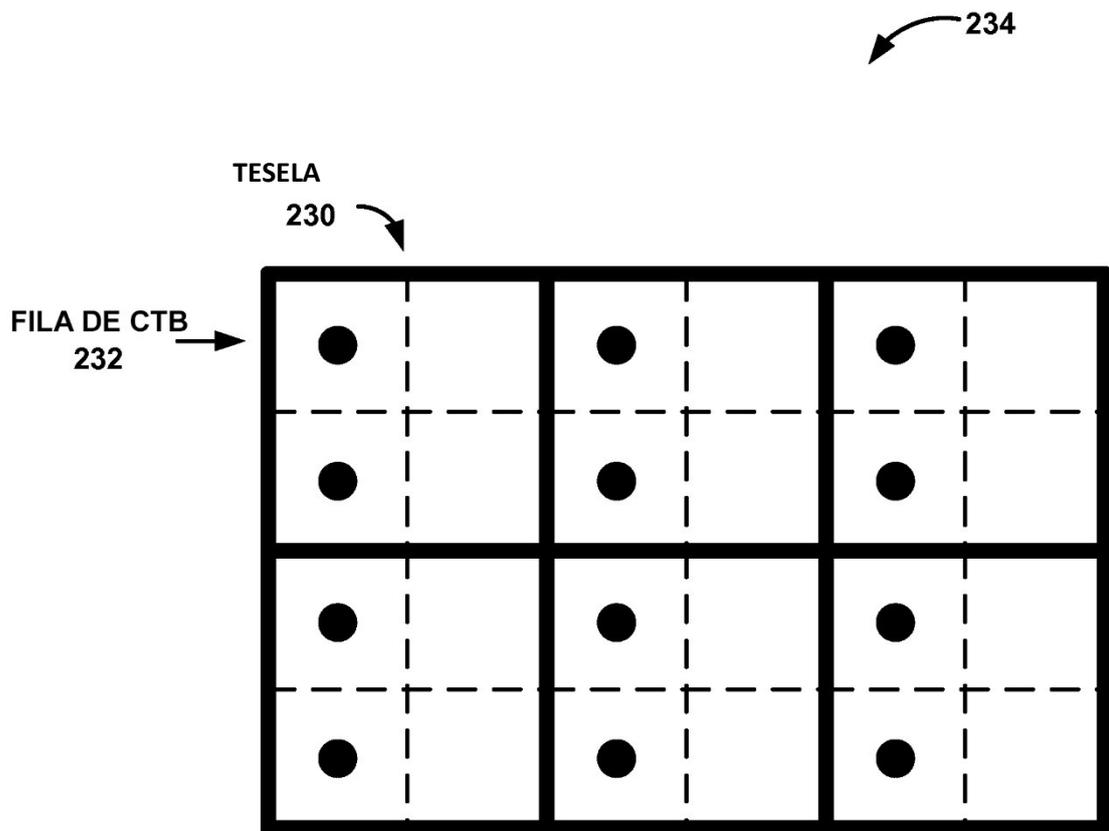


FIG. 7

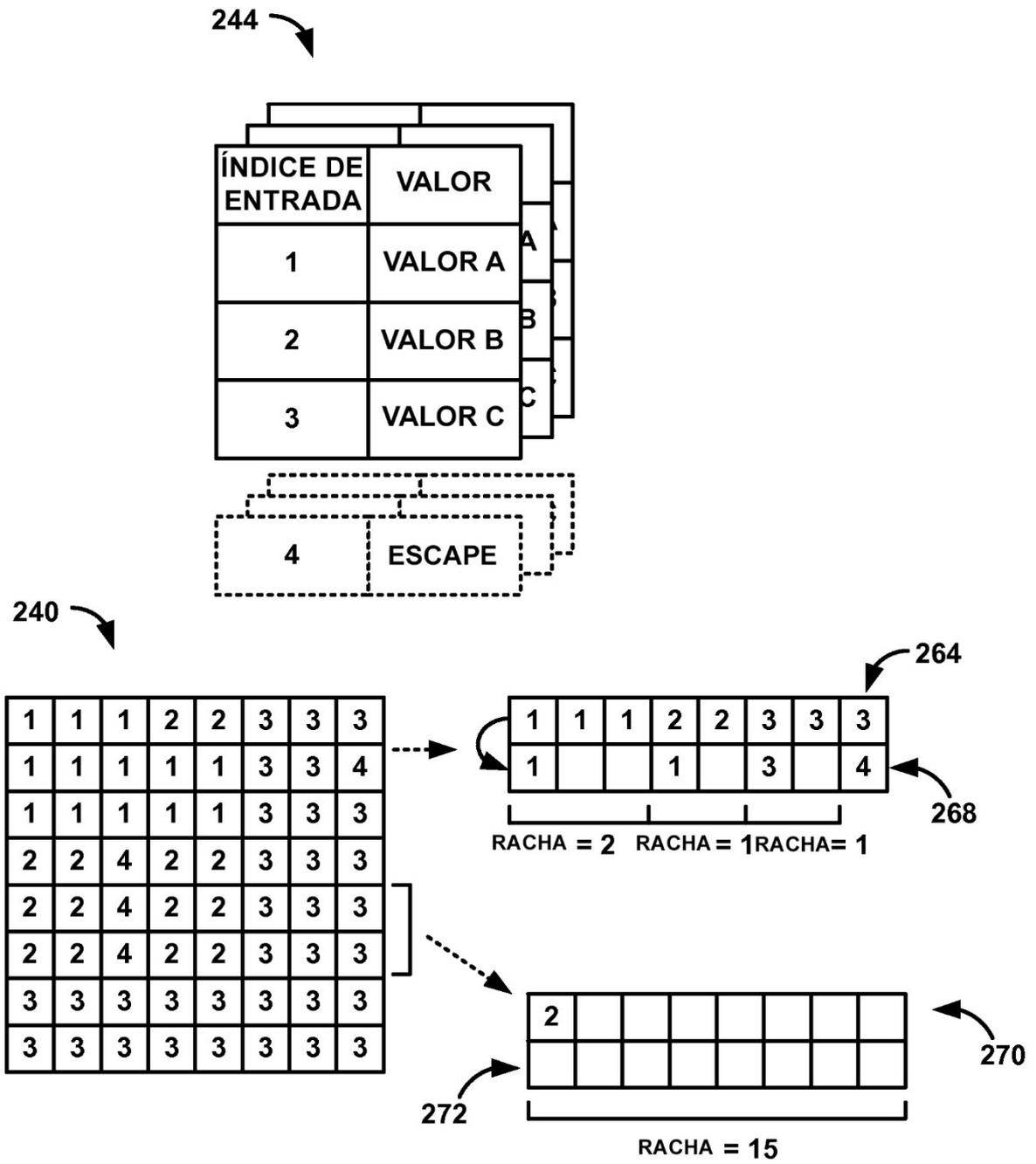


FIG. 8

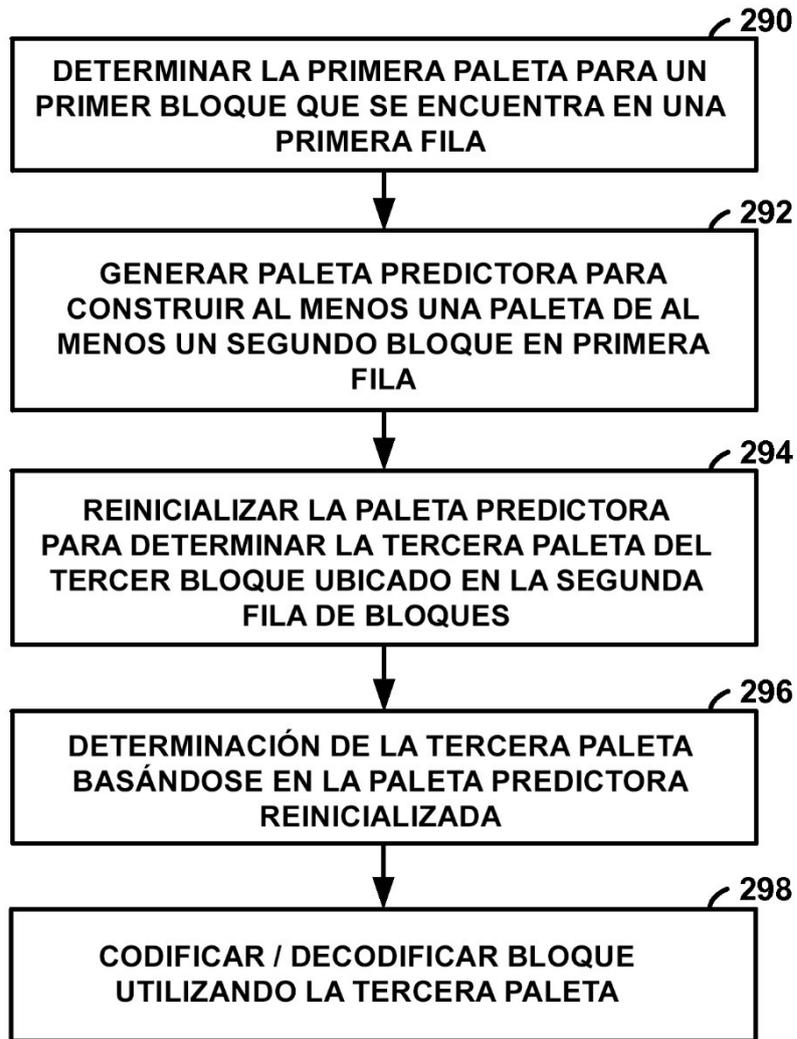


FIG. 9

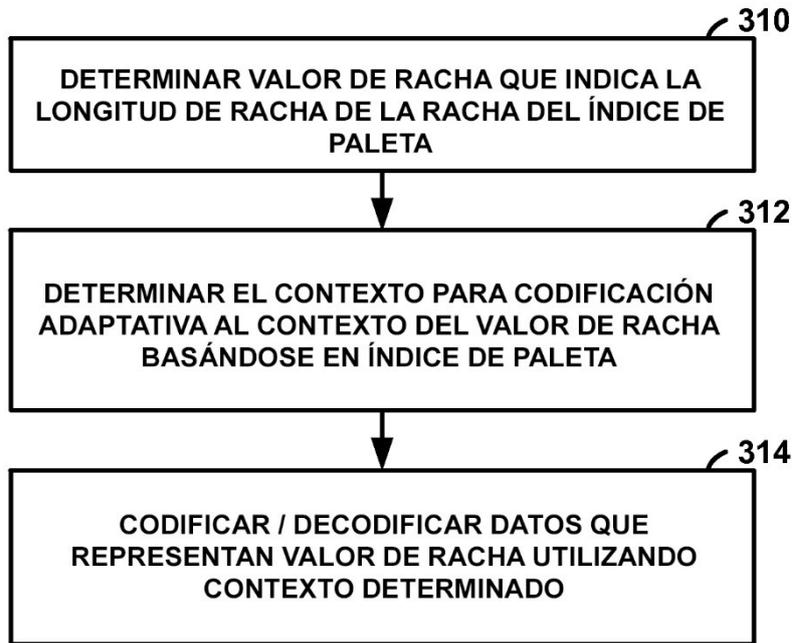


FIG. 10

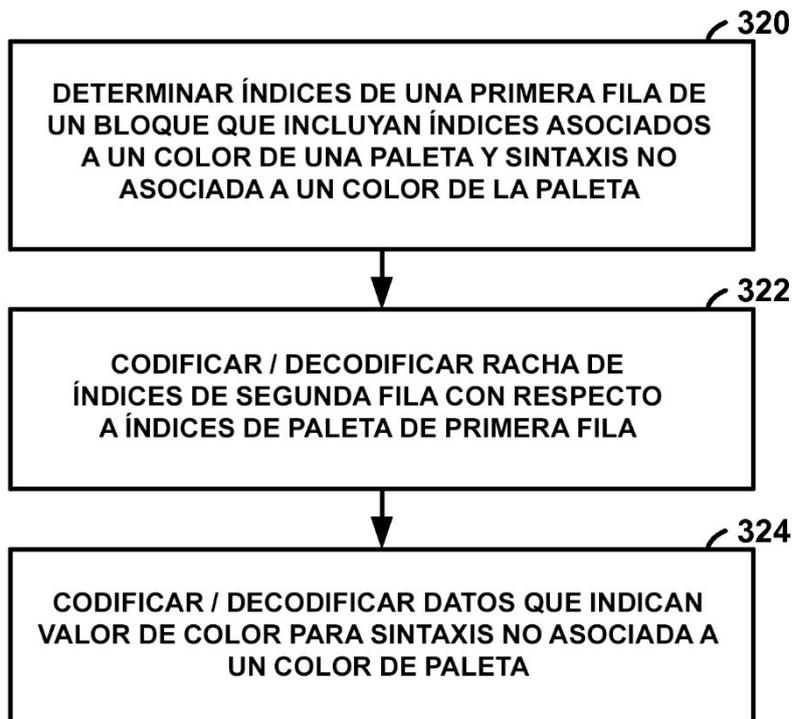


FIG. 11