

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 746 072**

51 Int. Cl.:

H04N 19/70 (2014.01)

H04N 19/463 (2014.01)

H04N 19/91 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **22.05.2015** **PCT/US2015/032250**

87 Fecha y número de publicación internacional: **26.11.2015** **WO15179804**

96 Fecha de presentación y número de la solicitud europea: **22.05.2015** **E 15729620 (3)**

97 Fecha y número de publicación de la concesión europea: **19.06.2019** **EP 3146722**

54 Título: **Parámetros de paletas máximos en codificación de vídeo basada en paletas**

30 Prioridad:

22.05.2014 US 201462002054 P
10.06.2014 US 201462010313 P
20.06.2014 US 201462015240 P
31.07.2014 US 201462031766 P
22.08.2014 US 201462040978 P
10.02.2015 US 201562114533 P
11.02.2015 US 201562115099 P
21.05.2015 US 201514719228

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
04.03.2020

73 Titular/es:

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

72 Inventor/es:

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

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 746 072 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

Parámetros de paletas máximos en codificación de vídeo basada en paletas

5 CAMPO TÉCNICO

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

ANTECEDENTES

10 [2] Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisores digitales, sistemas de radiodifusión directa digital, sistemas de radiodifusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, ordenadores de tableta, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, 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 Eficiencia (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 eficientemente, implementando dichas técnicas de compresión de vídeo.

25 [3] Las técnicas de compresión de vídeo realizan la predicción espacial (intra-imágenes) 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.

35 [4] 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 inter-codificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales que indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intra-codificado se codifica de acuerdo con un modo 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 transformada, dando como resultado coeficientes residuales, los cuales pueden cuantificarse posteriormente. Los coeficientes cuantificados, inicialmente dispuestos en una matriz bidimensional, pueden escanearse con el fin de producir un vector unidimensional de coeficientes, y puede aplicarse la codificación por entropía para lograr aún más compresión.

45 En el documento JCTVC-O0218 - GUO L ET AL: "Evaluation of Palette Mode Coding on HM-12.0+RExt-4.1 [Evaluación de codificación en modo de paletas en HM-12.0+RExt-4.1]", se discute la codificación basada en paletas para los contenidos de pantalla propuestos en JCTVC-N0249 de la 14ª reunión de JCTVC. Además, se considera la implementación de este procedimiento sobre el último software de Extensión de Rango HM-12.0+RExt-4.1. En el documento JCTVC-Q0037 - XIU X ET AL: "Description of screen content coding technology proposal by InterDigital [Descripción de propuesta de tecnología de codificación de contenido de pantalla]", se proporciona un procedimiento para la codificación de contenido de pantalla. Tanto la codificación mejorada de paletas como la conversión del espacio de color de los residuos de adaptación se analizan en función del marco actual de las extensiones de rango HEVC. Finalmente, en el documento JCTVC-Q0031 - CHEN J ET AL: "Description of screen content coding technology proposed by Qualcomm [Descripción de tecnología de codificación de contenido de pantalla propuesta por Qualcomm]", se detalla una propuesta en función de la propuesta 6 de extensiones de rango HEVC y el software HM13.0-RExt6.0. Los principales cambios considerados en el presente documento son extensiones de las herramientas actuales para intra BC, filtro de desbloqueo y RDPCM explícito, nuevas herramientas para paletas y transformación de color, y modificaciones de codificador y mejoras subjetivas para el contenido de la pantalla.

RESUMEN

60 [5] Las técnicas de esta divulgación se refieren a la codificación de vídeo basada en paletas. Por ejemplo, en la codificación basada en paletas, un codificador de vídeo (un codificador de vídeo o un decodificador de vídeo) puede formar una "paleta" como una tabla de colores para representar los datos de vídeo del área particular (por ejemplo, un bloque dado). La codificación basada en paletas puede ser especialmente útil para codificar áreas de datos de vídeo que tienen un número relativamente pequeño de colores. En lugar de codificar valores de píxeles reales (o sus residuos), el codificador de vídeo puede codificar índices de paletas para uno o más de los píxeles que relacionan los

píxeles con entradas en la paleta que representa los colores de los píxeles. Las técnicas descritas en esta divulgación también pueden incluir técnicas para diversas combinaciones de uno o más modos de codificación basadas en paletas de señalización, la transmisión de paletas, la predicción de paletas, y la transmisión de mapas de codificación basados en paletas y otros elementos sintácticos.

[6] En un ejemplo, un procedimiento para procesar datos de vídeo incluye determinar un valor de un elemento de sintaxis a nivel de bloque que indique, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque está codificada en función de un valor de color de al menos una muestra respectiva que no se incluye en una paleta de colores para codificar el bloque de datos de vídeo, y codificar el bloque de datos de vídeo en base al valor.

[7] En otro ejemplo, un dispositivo para procesar datos de vídeo incluye una memoria configurada para almacenar un bloque de muestras de datos de vídeo y uno o más procesadores configurados para determinar el valor de un elemento de sintaxis a nivel de bloque que indica, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque se codifica en función de un valor de color de la al menos una muestra respectiva que no esté incluida en una paleta de colores para codificar el bloque de datos de vídeo, y codificar el bloque de datos de vídeo basados en el valor.

[8] En otro ejemplo, un aparato para procesar datos de vídeo incluye medios para determinar un valor de un elemento de sintaxis a nivel de bloque que indica, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque está codificada en base a un valor de color de la al menos una muestra respectiva que no se incluye en una paleta de colores para codificar el bloque de datos de vídeo, y medios para codificar el bloque de datos de vídeo en función del valor.

[9] En otro ejemplo, un medio legible por ordenador no transitorio tiene almacenadas en el mismo instrucciones que, cuando se ejecutan, hacen que uno o más procesadores determinen el valor de un elemento de sintaxis a nivel de bloque que indica, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque se codifica en función de un valor de color de la al menos una muestra respectiva que no se incluye en una paleta de colores para codificar el bloque de datos de vídeo, y codificar el bloque de datos de vídeo según el valor.

[10] En otro ejemplo, un procedimiento para procesar datos de vídeo incluye la codificación de al menos uno de los datos que indica un tamaño máximo de paleta de una paleta de valores de color para codificar un bloque de datos de vídeo o datos que indican un tamaño máximo de predictor de paleta de un predictor de paleta para determinar la paleta de valores de color y codificar el bloque de datos de vídeo de acuerdo con los datos.

[11] En otro ejemplo, un dispositivo para procesar datos de vídeo incluye una memoria configurada para almacenar un bloque de datos de vídeo, y uno o más procesadores configurados para codificar al menos uno de los datos que indica un tamaño máximo de paleta de una paleta de valores de color para codificar el bloque de datos de vídeo o datos que indican un tamaño máximo de predictor de paleta de un predictor de paleta para determinar la paleta de valores de color, y codificar el bloque de datos de vídeo de acuerdo con los datos codificados desde el flujo de bits.

[12] En otro ejemplo, un aparato para procesar datos de vídeo incluye medios para codificar al menos uno de los datos que indica un tamaño máximo de paleta de una paleta de valores de color para codificar un bloque de datos de vídeo o datos que indican un tamaño máximo de predictor de paleta de un predictor de paleta para determinar la paleta de valores de color y medios para codificar el bloque de datos de vídeo de acuerdo con los datos.

[13] En otro ejemplo, un medio legible por ordenador no transitorio tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que uno o más procesadores codifiquen al menos uno de los datos que indica un tamaño máximo de paleta de una paleta de valores de color para codificar un bloque de datos de vídeo o datos que indican un tamaño máximo de predictor de paleta de un predictor de paleta para determinar la paleta de valores de color, y codificar el bloque de datos de vídeo de acuerdo con los datos.

[14] En otro ejemplo, un procedimiento para codificar datos de vídeo incluye determinar, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores utilizada para codificar el píxel, una longitud de ejecución de una ejecución de índices de paleta se codifican con el índice de paleta del píxel, determinar una longitud de ejecución máxima para una ejecución máxima de índices de paleta que se pueden codificar con el índice de paleta del píxel y datos de codificación que indican la longitud de ejecución en función de la longitud de ejecución máxima determinada.

[15] En otro ejemplo, un dispositivo para codificar datos de vídeo incluye una memoria configurada para almacenar un píxel de datos de vídeo asociados con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores utilizada para codificar el píxel, y uno o más procesadores configurados para determinar, para el píxel, una longitud de ejecución de una ejecución de índices de paleta que se codifican con el índice de paleta del píxel, que determina una longitud de ejecución máxima para una serie máxima de índices de paleta que se pueden codificar con el índice de paleta del píxel y datos de código que indican la longitud de ejecución en función de la longitud de ejecución máxima determinada.

[16] En otro ejemplo, un aparato para procesar datos de vídeo incluye medios para determinar, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores utilizada para codificar el píxel, una longitud de ejecución de una ejecución de índices de paleta se codifican con el índice de paleta del píxel, medios para determinar una longitud de ejecución máxima para una ejecución máxima de índices de paleta que se pueden codificar con el índice de paleta del píxel y medios para codificar datos que indican la longitud de ejecución en función de la longitud de ejecución máxima determinada.

[17] En otro ejemplo, un medio legible por ordenador no transitorio tiene instrucciones almacenadas en el mismo que, cuando se ejecutan, hacen que uno o más procesadores determinen, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores utilizada para codificar el píxel, una longitud de ejecución de una ejecución de índices de paleta que se codifican con el índice de paleta del píxel, determinar una longitud de ejecución máxima para una serie máxima de índices de paleta que se pueden codificar con el índice de paleta de píxel y datos de código que indican la longitud de ejecución en función de la longitud de ejecución máxima determinada.

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

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[19]

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

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

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

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

La FIG. 6 es un diagrama conceptual que ilustra un ejemplo de determinación de una longitud de serie máxima para un bloque de píxeles, coherente con las técnicas de esta divulgación.

La FIG. 7 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basado en uno o más elementos sintácticos a nivel de bloque que indican si cualquier muestra del bloque está codificada como muestras de salida, de acuerdo con las técnicas de esta divulgación.

La FIG. 8 es un diagrama de flujo que ilustra un proceso de ejemplo para decodificar un bloque de datos de vídeo basado en uno o más elementos sintácticos a nivel de bloque que indican si cualquier muestra del bloque se decodifica como una muestra de salida, coherente con las técnicas de esta divulgación.

La FIG. 9 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basado en uno o más elementos sintácticos que indican un tamaño de paleta máximo y un tamaño de predictor de paleta máximo, coherente con las técnicas de esta divulgación.

La FIG. 10 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basado en uno o más elementos sintácticos que indican un tamaño de paleta máximo y un tamaño de predictor de paleta máximo, coherente con las técnicas de esta divulgación.

La FIG. 11 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar (codificar o decodificar) datos que indican una longitud de serie de una serie de píxeles basada en una longitud de serie máxima potencial, coherente con las técnicas de esta divulgación.

DESCRIPCIÓN DETALLADA

[20] Los aspectos de esta divulgación están destinados a técnicas para la codificación y compresión de vídeo. En general, esta divulgación describe técnicas para la codificación de datos de vídeo basa en paleta. En la codificación

de vídeo tradicional, se supone que las imágenes son de tono continuo y espacialmente fluidas. Con base en estas suposiciones, se han desarrollado varias herramientas, como la transformada basada en bloques, el filtrado, etc., y dichas herramientas han demostrado un buen rendimiento para los vídeos de contenido natural.

[21] Sin embargo, en aplicaciones como el escritorio remoto, el trabajo colaborativo y la pantalla inalámbrica, el contenido de la pantalla generado por ordenador puede ser el contenido dominante a comprimir. Este tipo de contenido tiende a tener tonos discretos y líneas definidas, y límites de objetos de alto contraste. La suposición de tono continuo y suavidad puede que ya no se aplique y, por lo tanto, las técnicas tradicionales de codificación de vídeo pueden ser formas ineficaces de comprimir el contenido.

[22] Esta divulgación describe la codificación basada en paleta, que puede ser particularmente adecuada para la codificación de contenido generado por pantalla (por ejemplo, codificación de contenido de pantalla (SCC)). En algunos ejemplos, las técnicas para la codificación basada en paleta de datos de vídeo pueden usarse con una o más técnicas de codificación diferentes, tales como técnicas para codificación inter o intra-predictiva. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o descodificador, o codificador-descodificador combinado (códec), se puede configurar para realizar una codificación inter e intra-predictiva, así como una codificación basada en paleta.

[23] En algunos ejemplos, las técnicas de codificación basadas en paleta pueden configurarse para su uso con una o más normas de codificación de vídeo. Por ejemplo, la codificación de vídeo de alta eficiencia (HEVC) es una nueva norma de codificación de vídeo que está en proceso de elaboración por parte del Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos en Imágenes en Movimiento (MPEG) de JSO/IEC. Un borrador de memoria descriptiva textual de HEVC reciente se describe en Bross et al., "High Efficiency Video Coding (HEVC) Text Specification Draft 10 (for FDIS & Consent)" [Borrador 10 de memoria descriptiva textual de codificación de vídeo de alta eficiencia (HEVC) (para FDIS y consenso)], JCVCL1003_v13, 12.ª reunión de JCT-VC de ITU-T SG16 WP 3 e ISO/IEC JCT 1/SC 29/WG 11, 14 - 23 de enero de 2013 ("borrador de HEVC 10").

[24] Con respecto al marco de HEVC, como ejemplo, las técnicas de codificación basadas en paleta pueden configurarse para usarse como un modo de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basadas en paleta pueden configurarse para usarse como un modo PU en el marco de HEVC. En consecuencia, todos los procesos descritos a continuación descritos en el contexto de un modo CU pueden, de forma adicional o alternativa, aplicarse a PU. Sin embargo, estos ejemplos basados en HEVC no se deben considerar una restricción o limitación de las técnicas de codificación basadas en paleta descritas en el presente documento, ya que tales técnicas se pueden aplicar para funcionar independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación de paleta pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[25] En la codificación basada en paleta, se puede suponer que un área particular de datos de vídeo tiene un número relativamente pequeño de colores. Un codificador de vídeo (por ejemplo, un codificador de vídeo o un descodificador 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 con 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 apropiado en la paleta.

[26] 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 índices de paleta (también conocida como valores de índice de paleta) para los píxeles que relacionan el valor de píxel con la paleta. Un descodificador de vídeo puede obtener, a partir de un flujo de bits codificado, una paleta para un bloque, así como índices de paleta para los píxeles del bloque. El descodificador de vídeo puede relacionar los índices de paleta de los píxeles con las entradas de la paleta para reconstruir los valores de píxel del bloque. Los píxeles (y/o los índices de paleta relacionados que indican un valor de píxel) en general se pueden denominar muestras.

[27] Se supone que las muestras en el bloque se procesan (por ejemplo, se escanean) usando el orden de escaneado de trama horizontal. Por ejemplo, el codificador de vídeo puede convertir un bloque bidimensional de índices de paleta en una matriz unidimensional escaneando los índices de paleta usando un orden de escaneado de trama horizontal. Del mismo modo, el descodificador de vídeo puede reconstruir un bloque de índices de paleta utilizando el orden de escaneado de trama horizontal. Por consiguiente, esta divulgación puede referirse a una muestra anterior como una muestra que precede a la muestra que se está codificando actualmente en el bloque en el orden de exploración. Se debe tener en cuenta que los escaneados que no sean un escaneado de trama horizontal, como el orden de escaneado de trama vertical, también pueden ser aplicables. El ejemplo anterior pretende proporcionar una descripción general de la codificación basada en paleta.

[28] Una paleta típicamente incluye entradas numeradas por un índice y que representan los valores o intensidades del componente de color (por ejemplo, RGB, YUV o similar). Tanto un codificador de vídeo como un descodificador de vídeo determinan el número de entradas de paleta, los valores de los componentes de color para

cada entrada de paleta y el orden 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.

[29] En algunos ejemplos, se puede componer una paleta utilizando información de bloques codificados previamente. Es decir, una paleta puede contener entradas de paleta predichas a partir de la(s) paleta(s) utilizada(s) para codificar los bloques previos. Por ejemplo, como se describe en el documento de presentación de norma Wei Pu et al., "AHG10: Suggested Software for Palette Coding based on RExt6.0" [AHG10: Software sugerido para la codificación de paleta 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 de una paleta de predictores. Una paleta de predictores puede incluir entradas de paleta de bloques codificados previamente usando el modo de paleta u otras muestras reconstruidas. Para cada entrada en la paleta de predictores, se puede codificar un indicador binario para indicar si la entrada asociada con el indicador se copia a la paleta actual (por ejemplo, se indica con indicador = 1). La secuencia de indicadores binarios se puede denominar vector de predicción de paleta binaria. La paleta para codificar un bloque actual también puede incluir un 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 nuevas entradas puede indicar el tamaño total de la paleta en el bloque.

[30] Como se propone en JCTVC-Q0094, cada muestra en un bloque codificado con un modo de codificación basado en paleta se puede codificar usando uno de los tres modos de paleta, como se establece a continuación:

- Modo de salida: en este modo, el valor de muestra no se incluye en una paleta como una entrada de paleta y el valor de muestra cuantificado 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 cuantificados.
- Modo CopyFromTop (también denominado modo CopyAbove): en este modo, el índice de entrada de paleta para la muestra actual se copia desde la muestra ubicada directamente encima en un bloque.
- Modo de valor (también denominado modo índice): en este modo, el valor del índice de entrada de paleta se indica explícitamente.

[31] Como se describe en el presente documento, un índice de entrada de paleta se puede denominar un índice de paleta o simplemente 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 con 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 paleta se puede aplicar para codificar vídeo monocromo. Por consiguiente, el "valor de color" puede referirse en general a cualquier componente de color o no color utilizado para generar un valor de píxel.

[32] Para los modos CopyFromTop y de valor, también se puede señalar un valor de serie (que también puede denominarse simplemente serie). Un valor de serie puede indicar un número de muestras consecutivas (por ejemplo, una serie de muestras) en un orden de escaneado particular en un bloque codificado con paleta que están codificadas juntas. En algunos casos, la serie de muestras también se puede denominar una serie de índices de paleta, porque cada muestra de la serie tiene un índice asociado a una paleta.

[33] Un valor de serie puede indicar una serie de índices de paleta que están codificados usando el mismo modo de codificación de paleta. Por ejemplo, con respecto al modo 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 serie que indica un número de muestras consecutivas en un orden de escaneado que tienen el mismo índice de paleta y que están siendo codificadas con el índice de paleta. Con respecto al modo CopyFromTop, 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 contigua anterior (por ejemplo, una muestra que se coloca encima de la muestra codificada actualmente en un bloque) y un valor de serie que indica un número de muestras consecutivas en un orden de escaneado que también copia un índice de paleta de una muestra contigua superior y que se está codificando con el índice de paleta. En consecuencia, en los ejemplos anteriores, una serie de índices de paleta se refiere a una serie de índices de paleta que tienen el mismo valor o una serie de índices de paleta que se copian de los índices de paleta contiguos.

[34] Por lo tanto, la serie puede especificar, para un modo determinado, el número de muestras subsiguientes que pertenecen al mismo modo. En algunos casos, la señalización de un índice y un valor de serie puede ser similar a la codificación de longitud de serie. En un ejemplo con fines ilustrativos, una secuencia de índices de paleta consecutivos

de un bloque puede ser 0, 2, 2, 2, 5 (por ejemplo, cuando 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 índice de paleta de dos) usando el modo de valor. Después de codificar un índice que es igual a 2, el codificador de vídeo puede codificar una serie de tres, lo que indica que las tres muestras subsiguientes también tienen el mismo índice de paleta de dos. De manera similar, codificar una serie de cuatro índices de paleta después de codificar un índice usando el modo CopyFromTop puede indicar que se copian un total de cinco índices de paleta de los índices de paleta correspondientes en la fila sobre la posición de muestra que se está codificando actualmente.

[35] Las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de uno o más modos de codificación basados en paleta de señalización, paletas de transmisión, paletas de obtención y mapas de codificación basados en paleta de transmisión y otros elementos sintácticos. En algunos ejemplos, las técnicas de esta divulgación se pueden usar para resolver posibles redundancias asociadas con la señalización de los modos de paleta, índices de paleta, series y tamaños de paleta que están presentes en JCTVC-Q0094 (así como el software de referencia que implementa el modo paleta que se cargó con la contribución JCTVC-Q0094). En consecuencia, como se describe con mayor detalle a continuación, las técnicas de esta divulgación pueden, en algunos casos, mejorar la eficiencia y mejorar la velocidad de transferencia de bits cuando se codifican datos de vídeo usando un modo de paleta.

[36] Ciertos aspectos de esta divulgación están destinados a modos de codificación basados en paleta de señalización y, en particular, a técnicas asociadas con muestras de salida de señalización. Por ejemplo, las muestras de salida (también denominadas píxeles de salida) pueden ser muestras (o píxeles) de un bloque que no tiene un color correspondiente representado en una paleta para codificar el bloque. En consecuencia, las muestras de salida pueden no reconstruirse utilizando una entrada de color (o valor de píxel) de una paleta. En cambio, los valores de color para las muestras de salida se señalan en un flujo de bits por separado de los valores de color de la paleta.

[37] Como se describe con mayor detalle a continuación, un codificador de vídeo (por ejemplo, un codificador de vídeo y un decodificador de vídeo, pueden codificar datos por muestra que indiquen si una muestra de un bloque codificado con paleta está codificada basándose en el color de la muestra no incluida en una paleta para el bloque, por ejemplo, utilizando el proceso denominado "Modo de salida" anterior. En un ejemplo, el codificador de vídeo puede codificar un indicador para cada muestra que indique si la muestra está codificada como una muestra de salida, por ejemplo, usando el modo de salida (denominado en el presente documento señalización de salida implícita). En otro ejemplo, el codificador de vídeo puede codificar otra sintaxis (como un índice de paleta adicional, como se describe a continuación) para una muestra que indica que la muestra está codificada como una muestra de salida, por ejemplo, usando el modo de salida (denominado en el presente documento señalización de salida explícita).

[38] De acuerdo con los aspectos de esta divulgación, para un bloque codificado con paleta, uno o más elementos sintácticos pueden indicar, a nivel de bloque (por ejemplo, un nivel CU o nivel LCU), si alguna muestra del bloque está codificada basándose en un valor de color de la muestra que no se incluye en la paleta, por ejemplo, codificada como una muestra de salida. El uno o más elementos sintácticos pueden denominarse sintaxis de salida a nivel de bloque. Por ejemplo, la sintaxis a nivel de bloque puede referirse a la sintaxis codificada o determinada al codificar un bloque de datos de vídeo, como una CU o LCU. La sintaxis a nivel de bloque se puede incluir en una cabecera o con otros datos que están asociados con el bloque (por ejemplo, datos que se codifican antes o después de un bloque que describe una característica del bloque). Por el contrario, otra sintaxis que no sea sintaxis a nivel de bloque se puede incluir en una cabecera de fragmento o con píxeles individuales de datos de vídeo.

[39] En un ejemplo, un codificador de vídeo puede configurarse para codificar y/o determinar un indicador (que puede denominarse indicador de salida a nivel de bloque) que indica si alguna muestra del bloque está codificada basándose en un valor de color no incluido en la paleta. Por ejemplo, un valor de indicador de cero puede indicar que ninguna de las muestras del bloque está codificada usando el modo de salida. Es decir, el valor de todas las muestras de un bloque se puede determinar basándose en un valor de color que se incluye en una paleta para codificar el bloque. Un valor de indicador de uno puede indicar que al menos una muestra del bloque está codificada usando el modo de salida. Es decir, el valor de al menos una muestra no se incluye en una paleta para codificar el bloque y se puede señalar por separado. Por lo tanto, el indicador puede indicar, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra del bloque tiene un valor de color que no está incluido en una paleta para codificar el bloque.

[40] Como se describe en mayor detalle a continuación, la sintaxis de salida a nivel de bloque puede dar lugar, en algunos casos, a un ahorro de bits. Por ejemplo, al determinar si cualquier muestra de un bloque completo está codificada como una muestra de salida, el codificador de vídeo puede ser capaz de omitir la codificación de ciertos elementos sintácticos asociados con las muestras de salida. Es decir, en casos en que la sintaxis indica que no hay muestras codificadas como muestras de salida, el codificador de vídeo puede no codificar ninguna otra sintaxis asociada con muestras de salida para el bloque (por ejemplo, como la sintaxis por muestra indicada anteriormente). Como se describe con mayor detalle a continuación, el codificador de vídeo también puede omitir la codificación de cierta sintaxis cuando la sintaxis indica que al menos una muestra de un bloque está codificada como una muestra de salida basada en un tamaño de una paleta para el bloque que se codifica. En consecuencia, las técnicas de esta

divulgación pueden mejorar la velocidad de transferencia de bits y la eficiencia de codificación cuando se codifican datos de vídeo utilizando codificación basada en paleta.

[41] Otros aspectos de esta divulgación están destinados a la codificación de parámetros de paleta máximos para el modo de paleta. Por ejemplo, un tamaño de paleta máximo para una paleta puede ser típicamente un valor estático que se define tanto en un codificador de vídeo como en un descodificador de vídeo. Del mismo modo, un tamaño máximo de un predictor de paleta (usado para predecir paletas, como se describe en mayor detalle a continuación) también puede ser un valor estático que se define tanto en un codificador de vídeo como en un descodificador de vídeo. Por lo tanto, estos parámetros de paleta máximos no se pueden cambiar, independientemente de las características particulares de los datos de vídeo que se codifican.

[42] De acuerdo con los aspectos de esta divulgación, un codificador de vídeo puede configurarse para codificar datos que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo. Por ejemplo, de acuerdo con aspectos de esta divulgación, los datos que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo pueden incluirse en un conjunto de parámetros, tal como un conjunto de parámetros de secuencia (SPS). En consecuencia, el codificador de vídeo puede codificar al menos uno de los datos que indiquen un tamaño de paleta máximo de unos valores de paleta de colores para codificar un bloque de datos de vídeo o datos que indiquen un tamaño de predictor de paleta máximo de un predictor de paleta para determinar los valores de la paleta de colores.

[43] Los datos de codificación que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo pueden proporcionar flexibilidad, lo cual puede mejorar la eficiencia de la codificación. Por ejemplo, las técnicas pueden permitir que un codificador de vídeo use paletas y predictores de paleta de diferentes tamaños basándose en las características de los datos de vídeo que se codifican (por ejemplo, basándose en un bit de profundidad de los datos, un tamaño de bloque, un perfil o nivel asociado con los datos, o similares). De acuerdo con ello, los parámetros de paleta máximos se pueden adaptar a los datos de vídeo que se codifican, de modo que se pueden definir parámetros de paleta máximos relativamente mayores para los bloques que pueden beneficiarse de dichos parámetros. Además, se pueden definir parámetros de paleta máximos relativamente más pequeños para reducir la complejidad asociada con la construcción de paletas para bloques que tienen menos probabilidades de beneficiarse de los parámetros relativamente más grandes.

[44] Otros aspectos de esta divulgación están destinados a técnicas que codifican diversos elementos sintácticos para la codificación de vídeo basada en paleta. Por ejemplo, las técnicas de esta divulgación incluyen sintaxis de codificación para codificación de paleta, como un valor de serie (también denominado valor de longitud de serie) de índices de paleta, un vector de predicción de paleta u otra sintaxis relacionada con paleta, usando un código que considera un valor potencial máximo de la sintaxis que se codifica. En algunos casos, de acuerdo con aspectos de esta divulgación, la sintaxis puede codificarse usando una forma de código Golomb exponencial, como se describe en mayor detalle a continuación. Las técnicas pueden, en algunos casos, reducir la cantidad de bits necesarios para representar la sintaxis relacionada con la paleta.

[45] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo de ejemplo 10 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 descodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o la descodificación de vídeo. El codificador de vídeo 20 y el descodificador 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 con varios ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden estar configurados para codificar selectivamente varios bloques de datos de vídeo, tales como CU o PU en codificación HEVC, usando codificación basada en paleta o codificación no basada en paleta. Los modos de codificación no basados en paleta pueden referirse a diversos modos de codificación temporal inter-predictiva o modos de codificación espacial intra-predictiva, tales como los diversos modos de codificación especificados por borrador de HEVC 10.

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

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

[48] 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 comunicaciones que permiten al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. En este ejemplo, el dispositivo de origen 12 puede modular los datos de vídeo codificados de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y puede transmitir los datos de vídeo modulados al dispositivo de destino 14. El uno o más medios de comunicaciones pueden incluir medios de comunicaciones inalámbricos y/o cableados, tales como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El 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). El uno o más medios de comunicación pueden incluir routers, conmutadores, estaciones base u otros equipos que faciliten la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[49] 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, a través del acceso al disco o el acceso a la tarjeta. El medio de almacenamiento puede incluir una variedad de medios de almacenamiento de datos de acceso local, tales como discos Blu-ray, DVD, CD-ROM, memoria flash u otros medios adecuados de almacenamiento digital para almacenar datos de vídeo codificados.

[50] En un ejemplo adicional, el canal 16 puede incluir un servidor de archivos 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 archivos u otro dispositivo de almacenamiento intermedio a través de transmisión o descarga. El servidor de archivos puede ser un tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir los datos de vídeo codificados al dispositivo de destino 14. Entre los ejemplos de servidores de archivos se incluyen servidores web (por ejemplo, para un sitio web), servidores de protocolo de transferencia de archivos (FTP), dispositivos de almacenamiento conectados a la red (NAS) y unidades de disco local.

[51] El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de una conexión de datos estándar, como una conexión a Internet. Entre los ejemplos de tipos de conexiones de datos pueden incluirse canales inalámbricos (por ejemplo, conexiones Wi-Fi), conexiones alámbricas (por ejemplo, DSL, módem de cable, etc.), o una combinación de ambos que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el servidor de archivos puede ser una transmisión sin descarga, una transmisión de descarga o una combinación de ambas.

[52] 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 radiodifusiones de televisión por el aire, radiodifusiones de televisión por cable, transmisiones de televisión por satélite, transmisiones de vídeo en continuo, por ejemplo, mediante Internet, codificación de datos de vídeo para su almacenamiento en un medio de almacenamiento de datos, decodificación de datos de vídeo almacenados en un medio de almacenamiento de datos, u otras aplicaciones. En algunos ejemplos, el sistema de codificación de vídeo 10 puede configurarse para soportar transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como radiodifusión de vídeo, reproducción de vídeo, radiodifusión de vídeo y/o videotelefonía.

[53] 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 ajustes de codificación de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no incluyen necesariamente ninguna comunicación de datos entre dispositivos de codificación y decodificación. En otros ejemplos, los datos se recuperan de una memoria local, se transmiten 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 de 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 de la memoria.

[54] En el ejemplo de la FIG. 1, el dispositivo de origen 12 incluye una fuente de vídeo 18, un codificador de vídeo 20 y una interfaz de salida 22. En algunos ejemplos, la interfaz de salida 22 puede incluir un modulador/desmodulador (módem) y/o un transmisor. La fuente de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo 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 dichas fuentes de datos de vídeo.

[55] El codificador de vídeo 20 puede codificar datos de vídeo de la fuente de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 directamente transmite los datos de vídeo codificados al dispositivo destino 14 a través de un canal de comunicación 22. En otros ejemplos, los datos de vídeo codificados también pueden almacenarse en un

medio de almacenamiento o un servidor de archivos para un acceso posterior mediante el dispositivo de destino 14 para la decodificación y/o reproducción.

[56] 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 una variedad de dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

[57] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse como cualquiera entre una variedad de circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), conjuntos de puertas 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 de almacenamiento legible por ordenador no transitorio adecuado, y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cualquiera de 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.

[58] Esta divulgación puede referirse en general al codificador de vídeo 20 que "señala" o "transmite" cierta información a otro dispositivo, tal como el decodificador de vídeo 30. El término "señalar" o "transmitir" puede referirse en general a la comunicación de elementos sintácticos y/u otros datos usados para decodificar los datos de vídeo comprimidos. Dicha comunicación puede producirse en tiempo real o casi real. De forma alternativa, dicha comunicación puede producirse durante un lapso 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.

[59] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 funcionan de acuerdo con una norma de compresión de vídeo, tal como la norma HEVC mencionada anteriormente, y se describe en el borrador de HEVC 10. Además de la base norma HEVC, hay esfuerzos constantes para producir codificación de vídeo escalable, codificación de vídeo de múltiples visualizaciones y extensiones de codificación 3D para HEVC. Además, pueden proporcionarse modos de codificación basados en paleta, 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 paleta se pueden aplicar a codificadores y decodificadores configurados para funcionar de acuerdo con otras normas de codificación de vídeo, tales como la norma ITU-TH.264/AVC o normas futuras. En consecuencia, la aplicación de un modo de codificación basado en paleta para la codificación de unidades de codificación (CU) o unidades de predicción (PU) en un códec HEVC se describe con fines de ejemplo.

[60] En 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 matrices de muestras, denominadas S_L , S_{Cb} y S_{Cr} . S_L es una matriz bidimensional (es decir, un bloque) de muestras de luma. S_{Cb} es una matriz bidimensional de muestras de crominancia Cb. S_{Cr} es una matriz bidimensional de muestras de crominancia Cr. Las muestras de crominancia también se pueden denominar en el presente documento muestras de "croma". En otros casos, una imagen puede ser monocromática y puede solo incluir una matriz de muestras de luma.

[61] 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 bloques de árbol de codificación correspondientes de muestras de croma y estructuras sintácticas usadas para codificar las muestras de los bloques de árbol de codificación. Un bloque de árbol de codificación puede ser un bloque NxN de muestras. Una CTU también puede denominarse "bloque de árbol" o una "unidad de codificación más grande" (LCU). Las CTU de 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 por orden de codificación.

[62] 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 NxN de muestras. 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 unas matrices de muestra de luma, una matriz de muestra de Cb y una matriz de muestra 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 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.

[63] 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 el bloque de píxeles predictivos de una PU, el codificador de vídeo 20 puede generar los bloques de píxeles predictivos de la PU basándose en muestras descodificadas de la imagen asociada con la PU.

[64] 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 descodificadas de una o más imágenes distintas a la imagen asociada con 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 PU, la PU puede tener dos MV.

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

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

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

[68] Después de generar un bloque de coeficientes (por ejemplo, un bloque de coeficientes luma, un bloque de coeficientes Cb o un bloque de coeficientes Cr), el codificador de vídeo 20 puede cuantificar el bloque de coeficientes. La cuantificación se refiere en general a un proceso en el que los coeficientes de transformada se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes de transformada, proporcionando una compresión adicional. Después de que el codificador de vídeo 20 cuantifique un bloque de coeficientes, el codificador de vídeo 20 puede codificar por entropía elementos sintácticos que indican los coeficientes de transformada cuantificados. Por ejemplo, el codificador de vídeo 20 puede realizar Codificación Aritmética Binaria Adaptable al Contexto (CABAC) en los elementos sintácticos que indican los coeficientes de transformada cuantificados. El codificador de vídeo 20 puede emitir los elementos sintácticos codificados por entropía, en un flujo de bits.

[69] 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 NAL incluye una cabecera de la unidad NAL y encapsula una carga útil de secuencia de bytes sin procesar (RBSP). La cabecera de la unidad NAL puede incluir un elemento sintáctico que indica un código de tipo de unidad NAL. El código de tipo de unidad NAL especificado por la cabecera de unidad NAL de una unidad

NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contiene un número entero de bytes que se encapsula dentro de una unidad NAL. En algunos casos, una RBSP incluye cero bits.

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

[71] El descodificador de vídeo 30 puede recibir un flujo de bits generado por el codificador de vídeo 20. Además, el descodificador de vídeo 30 puede analizar el flujo de bits para descodificar elementos sintácticos del flujo de bits. El descodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo basándose al menos en parte en los elementos sintácticos descodificados del flujo de bits. El proceso para reconstruir los datos de vídeo puede ser en general recíproco al proceso realizado por el codificador de vídeo 20. Por ejemplo, el descodificador de vídeo 30 puede usar MV de PU para determinar bloques predictivos para las PU de una CU actual. Además, el descodificador de vídeo 30 puede cuantificar de forma inversa bloques de coeficientes de transformada cuantificados asociados con TU de la CU actual. El descodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes de transformada para reconstruir los bloques de transformada asociados con las TU de la CU actual. El descodificador de vídeo 30 puede reconstruir los bloques de codificación de la CU actual añadiendo las muestras de los bloques de predicción para las PU de la CU actual a muestras correspondientes de los bloques de transformada de las TU de la CU actual. Mediante la reconstrucción de los bloques de codificación para cada CU de una imagen, el descodificador de vídeo 30 puede reconstruir la imagen.

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

[73] 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 índices de paleta para los píxeles relacionados con el valor de píxel a la paleta. El descodificador 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 valores píxeles del bloque. El descodificador de vídeo 30 puede relacionar los índices de paleta de los píxeles con las entradas de la paleta para reconstruir los valores de píxel del bloque.

[74] Como se indicó anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar varios modos diferentes de codificación de paleta para codificar los índices de paleta de una paleta. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar un modo de salida, un modo CopyFromTop (también denominado modo Copy Above), o un modo de valor (también denominado modo Index) para codificar índices de paleta de un bloque. En general, codificar una muestra usando el "modo de salida" 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 indicó anteriormente, tales muestras pueden denominarse muestras de salida o píxeles de salida.

[75] Otro modo de codificación de paleta de ejemplo se describe en un tercer experimento de núcleo de codificación de contenido de pantalla, subprueba B.6, como se describe en Yu-Wen Huang et al., "Description of Screen Content Core Experiment 3 (SCCE3): Palette Mode" [Descripción del experimento de núcleo de contenido de pantalla 3 (SCCE3): Modo de paleta]. JCTVC-Q 1123, Valencia, ES, 27 de marzo - 4 de abril 2014 (en lo sucesivo, Q1123), otro modo se introdujo en el software lanzado por Canon el 26 de mayo de 2014. La macro para este modo era "CANON_NEW_RUN_LAST_TRANSITION" y se puede denominar en el presente documento modo de serie de transición. La serie de transición puede ser similar al modo de valor en ese codificador de vídeo 20 o el descodificador de vídeo 30 puede codificar un valor de índice seguido por una serie que especifica el número de muestras posteriores que tienen el mismo índice de paleta.

[76] La diferencia entre el modo de valor y el modo serie de transición es que el índice de paleta del modo de serie de transición no se señala en el flujo de bits. Por el contrario, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir el índice de paleta. Como se describe en el presente documento, inferir un valor puede referirse a la determinación de un valor sin referencia a la sintaxis dedicada que representa el valor que está codificado en un flujo de bits. Es decir, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir un valor sin codificar un elemento sintáctico dedicado para el valor en un flujo de bits. El índice inferido se puede denominar índice de transición.

[77] En algunos ejemplos, puede haber dos formas de señalar los modos de paleta. Una primera técnica para señalar modos de paleta se puede denominar señalización de salida explícita. Por ejemplo, en JCTVC-Q0094, si la macro "PLT_REMOVE_ESCAPE_FLAG" es cero, el codificador de vídeo 20 puede codificar explícitamente un indicador de salida para cada muestra de un bloque para indicar si una muestra que está codificada en un bloque está codificada en modo de salida. Si la muestra no está codificada con el modo de salida, el codificador de vídeo 20 puede codificar datos adicionales para indicar si el modo es CopyFromTop o de valor. En algunos casos, los datos adicionales pueden ser un indicador, denominado en el presente documento indicador de SPoint (por ejemplo, un valor de indicador SPoint de cero puede indicar el modo CopyFromTop y un valor de indicador SPoint de uno puede indicar el modo de valor, o viceversa).

[78] Por lo tanto, con la señalización de salida explícita, el indicador de SPoint puede usarse para indicar un tipo de serie particular para una serie de valores de píxel asociados con el modo indicado. Por ejemplo, el codificador de vídeo 20 puede codificar un indicador de SPoint para indicar si el índice que se está codificando actualmente y la serie de los índices de paletas posteriores que se codifican en una serie se codifican utilizando el modo CopyFromTop o de valor. El codificador de vídeo 20 no codifica el indicador de salida (por ejemplo, "PLT_REMOVE_ESCAPE_FLAG") y el indicador SPoint (cuando es necesario) para las muestras de serie posteriores. Es decir, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden inferir los valores del indicador de salida y el indicador de SPoint para las muestras incluidas en una serie. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden determinar el valor del indicador de salida y el indicador de SPoint para las muestras incluidas en la serie sin referencia a la sintaxis dedicada que representa dichos valores en el flujo de bits.

[79] Una segunda técnica para señalar modos de paleta se puede denominar señalización de salida implícita. Por ejemplo, si la macro "PLT_REMOVE_ESCAPE_FLAG" de JCTVC-Q0094 es una, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para aumentar el número de entradas de paleta de una paleta en una para adaptar un índice especial a la paleta que no corresponde a cualquier entrada de paleta. 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 del modo de salida.

[80] Al realizar una señalización de salida 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 como una muestra de salida (por ejemplo, una muestra que no tiene un valor de color representado en una paleta para codificar el bloque). El codificador de vídeo 20 también puede codificar el (los) valor(es) de color de la muestra de salida. En consecuencia, en el caso de la señalización de salida implícita, solo hay dos modos posibles (por ejemplo, modo CopyFromTop o modo de valor (también denominado modo de índice)) que se señalarán utilizando la sintaxis explícita. Por ejemplo, solo se puede señalar el indicador de SPoint para distinguir entre los modos. Si una muestra está codificada en modo de valor y el índice para el modo de valor es igual al índice de salida (por ejemplo, el índice adicional anotado anteriormente a la paleta), el codificador de vídeo 20 y el decodificador de vídeo 30 pueden inferir que la muestra se codificará como muestra de salida. En este caso, no se señala ninguna serie. Cuando se utiliza la señalización de salida implícita con el modo de serie de transición, el indicador SPoint puede tomar valores 0 (por ejemplo, modo de valor), 1 (por ejemplo, modo CopyFromTop) o 2 (por ejemplo, modo de serie de transición).

[81] Las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de uno o más modos de codificación basados en paleta de señalización, paletas de transmisión, paletas de obtención y mapas de codificación basados en paleta de transmisión y otros elementos sintácticos. En algunos ejemplos, las técnicas de esta divulgación se pueden usar para resolver posibles redundancias asociadas con la señalización de los modos de paleta, índices de paleta, series y tamaños de paleta que están presentes en JCTVC-Q0094 (así como el software de referencia que implementa el modo paleta que se cargó con la contribución JCTVC-Q0094).

[82] En el software asociado con las técnicas descritas en JCTVC-Q0094, ciertas redundancias de señalización ya se han considerado y eliminado. Por ejemplo, en JCTVC-Q0094, el indicador SPoint no se señala para muestras en la primera fila del bloque, porque un bloque codificado con el modo paleta no puede usar típicamente muestras reconstruidas de un bloque contiguo superior para predecir el bloque actual. Un bloque contiguo superior en general puede referirse a un bloque contiguo y que se encuentra encima de un bloque. De forma similar, si el modo para una muestra que precede a una muestra que se está codificando actualmente es CopyFromTop, el modo para el píxel actual no puede ser CopyFromTop.

[83] Sin embargo, esta divulgación reconoce otras redundancias y/o ineficiencias de señalización, que pueden eliminarse total o selectivamente. Como se describe en mayor detalle a continuación, las técnicas mejoran la eficiencia de la velocidad de transferencia de bits de codificación de vídeo sin afectar materialmente a la distorsión. Como un ejemplo, si la muestra directamente sobre una muestra actual es una muestra de salida, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para no codificar la muestra actual usando el modo CopyFromTop. En este caso, el codificador de vídeo 20 puede no señalar el SPoint para la muestra, y el codificador de vídeo 20 y el decodificador de vídeo 30 pueden inferir que el indicador de SPoint sea igual al modo de valor si es necesario.

[84] En otro ejemplo, de acuerdo con las técnicas de esta divulgación, si ni la muestra anterior ni la muestra directamente encima de la muestra actual en un bloque son muestras de salida y las muestras anteriores y superiores tienen el mismo índice de paleta, el codificador de vídeo 20 y el descodificador de vídeo 30 están configurados para no codificar la muestra actual utilizando el modo CopyFromTop. Esto se debe a que, para el modo CopyFromTop, el índice de la muestra actual sería el mismo que el de la muestra anterior. Si el modo para la muestra anterior era el modo de valor, la serie asociada con el modo de valor se extendería en uno para incorporar la muestra actual. Por otro lado, si el modo para la muestra anterior fue CopyFromTop, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para no codificar la muestra actual usando el modo CopyFromTop, como se indicó anteriormente. Por lo tanto, en este caso, el codificador de vídeo 20 puede no señalar el indicador SPoint para la muestra actual, y el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que el indicador SPoint sea igual al modo de valor si es necesario.

[85] En otro ejemplo, de acuerdo con las técnicas de esta divulgación, si la serie anterior es mayor o igual que el ancho del bloque codificado menos uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para no codificar la muestra actual usando el modo CopyFromTop. Dado que el modo CopyFromTop puede no seguir al modo CopyFromTop, como se describió anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que si el modo asociado con la muestra anterior se codifica utilizando el modo CopyFromTop, el modo de la muestra actual puede no codificarse usando el modo CopyFromTop. Si la serie anterior se codificó usando el modo de valor y la serie anterior fue mayor o igual que el ancho del bloque menos uno, el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden configurar para determinar que los índices de paleta para la muestra anterior y la muestra directamente encima de la muestra actual son las mismas (de manera similar al ejemplo descrito anteriormente). En este caso, si la muestra actual puede no tener el mismo índice, hace imposible el modo CopyFromTop. Por lo tanto, en este caso, el codificador de vídeo 20 puede no señalar el indicador SPoint para la muestra actual, y el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que el indicador SPoint sea igual al modo de valor si es necesario.

[86] En otro ejemplo, de acuerdo con los aspectos de esta divulgación, si el tamaño de paleta es uno para un bloque que se codifica cuando se usa señalización de salida explícita, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para no codificar ciertos índices de paleta, tales como los índices de paleta descritos en la Solicitud Provisional de Estados Unidos 61/845,824, presentada el 12 de julio de 2013, la Solicitud Provisional de Estados Unidos 61/899,048, presentada el 1 de noviembre de 2013, o la Solicitud Provisional de Estados Unidos 61/913,040, presentada el 6 de diciembre de 2013. Además, el codificador de vídeo 20 puede configurarse para no codificar el indicador SPoint, ya que el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que el indicador SPoint sea igual al modo de valor si es necesario. Esto es porque, si la muestra actual no está codificada como una muestra de salida (por ejemplo, una muestra que no tiene un valor de color representado en una paleta para codificar el bloque), el índice de paleta para la muestra actual ya es conocido y obtenido igual a cero (como el único índice de paleta posible). En este caso, solo se indica la serie. No es necesario distinguir entre los modos CopyFromTop y de valor, ya que ambos modos proporcionan un resultado idéntico. De forma similar, para la señalización de salida implícita, cuando el tamaño de la paleta es dos, el codificador de vídeo 20 puede señalar los índices de paleta para distinguir entre modos de valor y salida, pero la señalización del indicador SPoint no es necesaria por las mismas razones que antes.

[87] Las técnicas de esta divulgación también se pueden usar para eliminar redundancias cuando se usan los modos de valor, CopiaFinTop y transición. Por lo tanto, las técnicas pueden mejorar la eficiencia de velocidad de transferencia de bits de codificación de vídeo sin afectar materialmente a la distorsión. En un ejemplo con fines ilustrativos, una muestra actual se codifica en modo de valor y el modo serie de transición no está disponible para su uso (por ejemplo, solo están disponibles los modos CopyFromAbove y de valor). En este ejemplo, cuando el modo para la muestra anterior es valor, el índice de la muestra actual no puede ser el mismo que el de la muestra anterior; de lo contrario, la muestra actual se incluye en el modo de valor anterior y se incrementa en uno la serie para el modo de valor. De manera similar, cuando el modo para la muestra anterior es CopyFromTop, el índice de la muestra actual a codificar no puede ser el mismo que el anterior; de lo contrario, la muestra actual se codificará con el modo CopyFromTop y posiblemente la serie para el modo CopyFromTop se incrementará en uno.

[88] Teniendo en cuenta la relación descrita anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden reducir el índice de la muestra actual en uno cuando el índice es mayor que el índice de la muestra anterior (por ejemplo, si la muestra anterior está en modo de valor) o la muestra superior (por ejemplo, si la muestra anterior está en modo CopyFromTop). Este proceso se describe en C. Gisquet et al., "AHG10: Palette Index Coding" [AHG10: Codificación del índice de paleta], JCTVC-Q0064, Valencia, ES, del 27 de marzo al 4 de abril de 2014 (en adelante, JCTVC-Q0064). Además, el número máximo de índices de paleta posible se puede reducir en uno, independientemente de si la condición anterior es verdadera (el índice actual es mayor que los índices anteriores de la paleta izquierda o superior). Por ejemplo, cuando se usa un código de longitud variable (por ejemplo, como un código binario truncado) para codificar el índice, el número de entradas de paleta puede reducirse en uno.

[89] De acuerdo con aspectos de esta divulgación, de forma alternativa o adicionalmente al proceso descrito anteriormente, el proceso de ajuste del índice en modo de valor puede modificarse adicionalmente cuando se usa el modo de serie de Transición. Por ejemplo, de acuerdo con los aspectos de esta divulgación, si la muestra no es la

primera muestra en el bloque y la muestra anterior no está codificada como muestra de salida, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden realizar el proceso de ajuste de índice descrito a continuación.

[90] De acuerdo con aspectos de esta divulgación, si la muestra anterior está codificada en modo de valor o de serie de Transición, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el número de entradas de paleta en una. Además, si el valor de índice es mayor o igual que el valor de índice de la muestra anterior (índice o índice de transición), el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el valor del índice actual en uno. Esta divulgación puede referirse a este valor reducido como el valor del índice ajustado. Entonces, si el valor de índice para la muestra anterior no es igual al índice de transición y el número de entradas de paleta es mayor que uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para establecer una variable "actualización" en uno, de lo contrario establecer la "actualización" a cero. Si la actualización es igual a uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden disminuir adicionalmente el número de entradas de paleta en una. Esta divulgación puede referirse al número reducido de entradas de paleta como el tamaño de paleta ajustado.

[91] Además, si el índice de transición es mayor o igual que el valor de índice para la muestra anterior, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el índice de transición en uno. Esta divulgación puede referirse al índice de transición reducido como el valor del índice de transición ajustado. Si la actualización es igual a uno y el valor del índice ajustado es mayor que el valor del índice de transición ajustado, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir adicionalmente el valor del índice ajustado en uno. Además, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para realizar el último ajuste del índice solo si el tamaño de paleta ajustado es mayor que uno. Esto se debe a que el valor del índice ajustado solo se puede señalar si el tamaño de paleta ajustado es mayor que uno.

[92] Si el tamaño de paleta ajustado es mayor que uno, el codificador de vídeo 20 puede codificar una indicación del valor de índice ajustado, teniendo en cuenta que el número máximo posible de índices de paleta puede ser igual al tamaño de paleta ajustado. En este caso, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar una binarización truncada, para la codificación, tal como la codificación binaria truncada descrita en el presente documento.

[93] En algunos ejemplos, de acuerdo con aspectos de esta divulgación, se puede llevar a cabo un proceso similar al proceso descrito anteriormente comprobando el valor de píxel y el modo utilizado para el píxel directamente encima de la muestra actual. Es decir, el proceso descrito anteriormente con respecto a un píxel posicionado a la izquierda del píxel actual puede realizarse en cambio para los píxeles contiguos superiores, donde el valor de muestra izquierdo y el modo descrito anteriormente se reemplazan por el valor y el modo de píxeles anteriores.

[94] Por ejemplo, si la muestra no está en la primera fila y la muestra anterior está codificada en modo CopyFromTop y la muestra anterior no está codificada como una muestra de salida, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir en uno el número de entradas de paleta. Además, si el valor de índice actual es mayor o igual que el valor de índice de la muestra directamente anterior, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el valor del índice actual en uno. Nuevamente, este valor de índice reducido se puede denominar el valor de índice ajustado. Entonces, si el valor de índice para la muestra directamente superior no es igual al índice de transición y el número de entradas de paleta es mayor que uno, el codificador de vídeo 20 puede establecer una actualización de variable en uno, o de lo contrario establecer la actualización en cero.

[95] Si la actualización es igual a uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir adicionalmente el número de entradas de paleta en una, lo cual puede denominarse el tamaño de paleta ajustado. Además, si el índice de transición es mayor o igual que el valor de índice para la muestra anterior, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el índice de transición en uno, que puede denominarse valor de índice de transición ajustado. Si la actualización es igual a cero y el valor del índice ajustado es mayor que el valor del índice de transición ajustado, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para disminuir el valor del índice ajustado en uno. Además, el último ajuste del índice puede realizarse solo si el tamaño de paleta ajustado es mayor que uno, porque el valor del índice ajustado típicamente solo se señala si el tamaño de paleta ajustado es mayor que uno.

[96] Si el tamaño de paleta ajustado es mayor que uno, el codificador de vídeo 20 puede configurarse para codificar una indicación del valor de índice ajustado, y puede, en algunos ejemplos, tener en cuenta el número máximo posible de índices de paleta igual al tamaño de paleta ajustado. En este caso, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar la binarización truncada, tal como la codificación binaria truncada descrita en el presente documento.

[97] La eliminación de redundancia en la señalización del índice de paleta en relación con el modo de serie de transición se describe más arriba. Sin embargo, estas técnicas pueden combinarse con un procedimiento de serie limitada, como se describe a continuación y en la Solicitud Provisional de Estados Unidos n.º 62/002,717 presentada el 23 de mayo de 2014 y la Solicitud Provisional de Estados Unidos n.º 62/009,772, presentada el 9 de junio de 2014.

En este caso, por encima de un cierto valor de índice de paleta, la serie es siempre igual a cero y, por lo tanto, para esos índices de paleta, el codificador de vídeo 20 puede configurarse para no codificar una indicación del valor de serie. Más bien, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para obtener el valor de serie para que sea igual a cero. Para esta combinación, las técnicas descritas anteriormente con respecto al modo de serie de transición permanecen sin cambios. Es decir, por ejemplo, las técnicas de eliminación de redundancia descritas anteriormente también se pueden usar con el modo de serie limitada.

[98] De forma adicional o alternativa, las técnicas de esta divulgación también pueden combinarse con la técnica de serie limitada tal como se propone en el documento de presentación de norma Guillaume Laroche et al., "AHG10: Run Coding for Palette Mode" [AHG10: Ejecutar la codificación para el modo de paleta], JCTVC-Q0066, Valencia, ES, 27 de marzo - 4 de abril de 2014 (en adelante, JCTVC-Q0066). En este ejemplo, también se especifica un índice de límite. Sin embargo, una diferencia con la técnica de serie limitada descrita anteriormente es que los índices de paleta mayores que un índice de límite también pueden tener series de uno o más. Sin embargo, el codificador de vídeo 20 puede no señalar las series. Al implementar esta segunda técnica de serie limitada, las técnicas de eliminación de redundancia de esta divulgación solo pueden aplicarse si el valor del índice del píxel anterior es menor o igual que el valor del índice límite, o el valor del índice del píxel anterior es menor que o igual al valor del índice límite.

[99] Las técnicas descritas anteriormente se describen en general con respecto a un codificador de vídeo (tal como un codificador de vídeo 20). En el lado del descodificador (como se implementa, por ejemplo, mediante el descodificador de vídeo 30), usando las mismas condiciones que en el lado del codificador, el descodificador de vídeo 30 también puede ajustar el número de entradas de paleta y el índice de transición. El descodificador de vídeo 30 puede descodificar el índice usando el número ajustado de entradas de paleta. El índice descodificado puede incrementarse (en lugar de reducirse) usando las mismas condiciones que en el lado del codificador.

[100] Ciertas técnicas descritas reducen la redundancia en casos en los que el modo CopyFromTop no es posible, y por lo tanto la señalización del indicador de SPoint puede modificarse de manera que el codificador de vídeo 20 y el descodificador de vídeo 30 puedan inferir el modo de valor. De acuerdo con aspectos de esta divulgación, las técnicas de reducción de redundancia pueden extenderse al caso en el que también se usa el modo de serie de transición. En este caso, el modo CopyFromTop no es posible si cualquiera de las siguientes condiciones es verdadera:

1. La muestra está en la primera fila.
2. El modo para la muestra anterior es CopyFromTop.
3. El píxel anterior está codificado en el modo de salida y la muestra no está en la primera fila y la muestra anterior no está codificada en el modo CopyFromTop.
4. La muestra anterior y la muestra anterior tienen el mismo índice y la muestra anterior no está codificada en el modo de salida.

[101] Las técnicas de esta divulgación también proporcionan una alternativa para señalar explícitamente un indicador de salida para el modo de paleta de salida. Por ejemplo, en lugar de señalar el indicador de salida antes del indicador SPoint con la señalización de salida explícita, de acuerdo con aspectos de esta divulgación, el orden de los indicadores puede intercambiarse al tiempo que también cambia la semántica de dichos indicadores. En este caso, el codificador de vídeo 20 puede señalar el indicador de SPoint primero en un flujo de bits. En este ejemplo, un indicador de SPoint que es igual a uno puede indicar el modo de valor, mientras que un indicador de SPoint que es igual a cero puede indicar que el modo de paleta para una muestra actual es CopyFromTop o Salida. Además, cuando el indicador de SPoint es igual a uno, el codificador de vídeo 20 puede indicar un indicador de salida para diferenciar entre el modo CopyFromTop y el modo de salida.

[102] En los ejemplos anteriores, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar CABAC para codificar al menos uno de los indicadores descritos anteriormente o ambos indicadores descritos anteriormente (por ejemplo, el indicador SPoint o el indicador de salida). De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar tales indicadores usando el modo de obtención CABAC para reducir el número de bins codificados por contexto.

[103] Como se describió anteriormente, el modo CopyFromTop puede no ser posible bajo ciertas condiciones. En tales casos, cuando se usa un procedimiento de señalización alternativo (por ejemplo, como el intercambio de indicadores), el codificador de vídeo 20 puede configurarse para que solo señale el indicador de SPoint sin señalar el indicador de salida. En este caso, el indicador SPoint puede tener una semántica diferente. Por ejemplo, un indicador de SPoint que es igual a uno puede indicar que el modo es el modo de valor, pero un indicador de SPoint que es igual a cero puede indicar un modo de Salida. Si el indicador SPoint está codificado por contexto utilizando CABAC, se puede usar un contexto independiente adicional para codificar el valor del indicador SPoint en los casos en que el modo CopyFromTop sea imposible. En el caso de que un tamaño de paleta sea uno y se use un modo de salida, como se describió anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para omitir la codificación del indicador SPoint cuando se utiliza el procedimiento de señalización alternativo.

[104] Las técnicas de esta divulgación también proporcionan otra técnica de señalización alternativa (por ejemplo, con respecto a JCTVC-Q0094) para señalar un indicador de salida para el modo de paleta de salida. Por ejemplo, en

JCTVC-Q0094, ciertas redundancias de señalización se han considerado y eliminado en el software de referencia. Como un ejemplo, cuando se codifica la muestra actual, si un modo de paleta para una muestra anterior es CopyFromTop, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden no codificar el píxel actual usando el modo CopyFromTop. De manera similar, si el modo para la muestra anterior es el modo de valor con índice de paleta "X", el codificador de vídeo 20 y el descodificador de vídeo 30 pueden no codificar el píxel actual usando el modo de valor con el mismo índice de paleta "X". En la etapa de análisis (por ejemplo, al analizar elementos sintácticos a partir de un flujo de bits codificado en el descodificador de vídeo 30), el descodificador de vídeo 30 verifica las condiciones indicadas anteriormente para determinar qué elementos sintácticos están permitidos para leer el flujo de bits correctamente. Este proceso de verificación puede volverse oneroso si se deben verificar muchas de esas condiciones.

[105] De acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para "reutilizar" las redundancias mencionadas anteriormente para señalar implícitamente el modo de salida. Por ejemplo, cuando se codifica una muestra actual de un bloque, si una muestra anterior se codifica utilizando el modo CopyFromTop y el modo para el píxel actual también se señala como CopyFromTop, el descodificador de vídeo 30 puede inferir que el modo para el bloque actual es el modo de salida. Es decir, el codificador de vídeo 20 puede usar la redundancia de dos muestras en una fila que se codifica con el modo CopyFromTop para señalar el modo de salida. De manera similar, si el modo para la muestra anterior a una muestra que actualmente se codifica es el modo de valor con el índice de paleta "X" y el modo señalado es el modo de valor con el mismo índice de paleta "X", el descodificador de vídeo 30 puede inferir el modo para el bloque actual para ser el modo de salida. De manera similar, otras redundancias descritas anteriormente también pueden aprovecharse de esta manera.

[106] En los ejemplos descritos anteriormente, el modo de salida de señalización basado en redundancias no incluye todas las situaciones posibles en las que el codificador de vídeo 20 puede señalar el modo de salida. En consecuencia, estas técnicas se pueden usar como una forma complementaria para señalar el modo de salida. En otros ejemplos, las técnicas se pueden imponer en el flujo de bits, de manera que el modo de salida solo se puede señalar en estas situaciones restringidas.

[107] Las técnicas de esta divulgación también se refieren a la señalización de que una muestra es una muestra de salida para un tamaño de paleta que es igual a cero. Por ejemplo, de acuerdo con los aspectos de esta divulgación, si un tamaño de paleta de una paleta asociada con un bloque que se codifica actualmente es igual a cero cuando se utiliza la señalización de salida explícita, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para inferir que todas las muestras en el bloque están codificadas como muestras de salida. Es decir, el codificador de vídeo 20 y el descodificador 30 pueden configurarse para determinar que todas las muestras en el bloque están codificadas como muestras de salida (por ejemplo, muestras que no tienen un valor de color representado en una paleta para codificar el bloque) sin codificación o descodificación de sintaxis dedicada que representa el modo de salida en un flujo de bits. Del mismo modo, si un tamaño de paleta de una paleta asociada con un bloque codificado actualmente es igual a uno cuando se utiliza señalización de salida implícita (por ejemplo, el único índice de la paleta es el índice adicional utilizado para señalar el modo de salida, como se describió anteriormente) el codificador 20 y el descodificador de vídeo 30 pueden configurarse para inferir que todas las muestras en el bloque están codificadas como muestras de salida.

[108] En ambos ejemplos descritos anteriormente (por ejemplo, tanto para la señalización de salida explícita como implícita), el codificador de vídeo 20 y el descodificador de vídeo 30 pueden omitir la codificación de cierta sintaxis basada en paleta para el resto del bloque. Por ejemplo, para la señalización de salida explícita, el codificador de vídeo 20 puede no señalar un indicador de salida para muestras del bloque. Además, tanto para la señalización de salida explícita como implícita, el codificador de vídeo 20 puede no señalar el indicador SPoint (para la señalización de salida tanto implícita como explícita). Es decir, dado que todas las muestras para el bloque pueden inferirse como muestras de salida, el codificador de vídeo 20 no necesita señalar al indicador Spoint para distinguir entre los modos CopyFromTop y de valor. El descodificador de vídeo 30 puede omitir también la descodificación de tal sintaxis, lo que puede mejorar la velocidad de transferencia de bits y la eficiencia de codificación.

[109] En un ejemplo alternativo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden restringir el tamaño de la paleta para que sea al menos uno de manera normativa. En este ejemplo, el codificador de vídeo 20 puede configurarse para modificar la señalización del tamaño de la paleta para que se señale (tamaño de la paleta - 1). Por ejemplo, cuando se usa un predictor de paleta (por ejemplo, como se describe con mayor detalle con respecto al ejemplo de la Fig. 4), para cada entrada del predictor de paleta, el codificador de vídeo 20 puede codificar un indicador de un bit para indicar si las entradas de predictor de paleta respectivas se incluyen en una paleta de un bloque actual. Estas entradas se conocen como las entradas de paleta predichas y se indican mediante un vector binario de predicción de paleta (por ejemplo, la secuencia de indicadores de un bit). El codificador de vídeo 20 también puede señalar el número de nuevas entradas de paleta después de las entradas predichas. En otros ejemplos, el codificador de vídeo 20 puede señalar el número de nuevas entradas de paleta antes de las entradas predichas. En cualquier caso, si el número de entradas de paleta predichas es cero, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indiquen (número de nuevas entradas de paleta --- 1) en lugar de codificar el número de nuevas entradas de paleta.

[110] En otro ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para restringir el modo de paleta de modo que un tamaño de paleta no sea igual a 0. Por ejemplo, esta restricción puede lograrse como una restricción de flujo de bits, es decir, un flujo de bits no puede contener un bloque codificado de paleta con un tamaño de paleta igual a cero.

[111] Las técnicas de esta divulgación también se refieren al tamaño de la paleta de señalización. Por ejemplo, el tamaño de la paleta para un bloque actual (por ejemplo, una CU actualmente codificada por el codificador de vídeo 20 o el descodificador de vídeo 30) puede señalarse explícitamente (por ejemplo, como se describe, por ejemplo, en la Solicitud de los Estados Unidos n.º 14/244,688, presentada el 3 de abril de 2014 y Solicitud de Estados Unidos n.º 14/244,711, presentada el 3 de abril de 2014). En dichos ejemplos, el tamaño de la paleta incluye tanto entradas de paleta predichas (por ejemplo, determinadas usando un predictor de paleta) como nuevas entradas de paleta (por ejemplo, como se señala explícitamente en el flujo de bits).

[112] De acuerdo con los aspectos de esta divulgación, si se señala el tamaño de la paleta, puede no ser necesario indicar el número de entradas nuevas, ya que el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para obtener el número de nuevas entradas de paleta para un bloque a partir del número de entradas predichas y el tamaño de la paleta (por ejemplo, tamaño de la paleta - número de entradas predichas = número de entradas nuevas). Además, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para terminar la predicción de entradas de paletas anteriores cuando se señala el tamaño de la paleta y se alcanza ese número de tamaño señalado al construir la paleta para un bloque actual.

[113] En algunos ejemplos, el tamaño de la paleta puede predecirse a partir de paletas anteriores, y el codificador de vídeo 20 puede configurarse para señalar solamente la diferencia. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar la diferencia entre el tamaño de la paleta y el tamaño de paleta predicho para un bloque usando un código Golomb exponencial, unario truncado o de longitud fija. En algunos casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para hacer que la predicción dependa (por ejemplo, esté basada en el) tamaño de bloque que se codifica. Por ejemplo, para un bloque de 8x8, el tamaño de la paleta puede predecirse desde la paleta asociada con el último bloque 8x8 codificado usando el modo de paleta (por ejemplo, el 8x8 más recientemente codificado en orden de exploración antes del bloque actual). Asimismo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para predecir un tamaño de paleta para un bloque de 16x16 basándose en una paleta de un bloque 16x16 previamente codificado, y una relación similar puede extenderse a bloques de otros tamaños. De forma alternativa, en otro ejemplo, se puede predecir un tamaño de paleta a partir del último bloque codificado con un tamaño menor o igual al tamaño del bloque actual.

[114] Las técnicas de esta divulgación también se refieren a la señalización de un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo. En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar tales datos desde un SPS. Los datos de codificación que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo pueden proporcionar flexibilidad, por ejemplo, permitiendo que el codificador de vídeo 20 y el descodificador de vídeo 30 usen paletas y predictores de paleta de diferentes tamaños para diferentes perfiles, niveles, profundidades de bits, tamaños de bloques, o similar. En el contexto de una norma de codificación de vídeo, un "perfil" corresponde a un subconjunto de algoritmos, características o herramientas y restricciones que se les aplican. Por ejemplo, un perfil puede ser un subconjunto de una sintaxis de flujo de bits completa que es especificada por un particular. Un nivel puede corresponder a las limitaciones del consumo de recursos del descodificador, tales como, por ejemplo, memoria de descodificador y cálculo, que están relacionadas con la resolución de las imágenes, velocidad de transferencia de bits y la velocidad de procesamiento de bloques. Un perfil puede ser señalizado con un valor de `idc_perfil` (indicador de perfil), mientras que un nivel puede ser señalizado con un valor de `idc_nivel` (indicador de nivel).

[115] De acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar información con respecto a un tamaño de paleta máximo para determinar los elementos e indicadores asociados con el modo de paleta, por ejemplo, para señalar el número de nuevas entradas de paleta. Como ejemplo, un tamaño máximo posible de la paleta se puede denotar como `MAX_PLT_SIZE`, que puede codificarse mediante el codificador de vídeo 20 y descodificarse mediante el descodificador de vídeo 30. De manera similar, un tamaño máximo posible de un vector de predicción de paleta se puede denotar como `MAX_PLT_PREDICTOR_SIZE`, que puede codificarse mediante el codificador de vídeo 20 y el descodificarse mediante el descodificador de vídeo 30.

[116] Como otro ejemplo, de acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador 30 pueden codificar datos que indican el número de "unos" en un vector binario de predicción de paleta (por ejemplo, que puede representar el número de entradas del predictor de paleta copiado a una paleta para codificar un bloque actual). En algunos casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un elemento sintáctico `numPredPalette` para indicar el número de las entradas de paleta predichas. Si el valor de `numPredPalette` es igual al valor de `MAX_PLT_SIZE` (es decir, el tamaño de paleta máximo), el codificador

de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para omitir la codificación del número de nuevas entradas de paleta. De lo contrario, si el valor de numPredPalette es menor que el valor de MAX_PLT_SIZE, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar una binarización truncada basándose en (MAX_PLT_SIZE - numPredPalette), que es el valor máximo posible para el número de nuevas entradas de paleta, para codificar los datos que indican el número de nuevas entradas.

[117] En general, la binarización truncada puede incluir cualquier técnica que utilice información sobre un valor máximo posible de un parámetro particular que se señala (por ejemplo, como el número de nuevas entradas de paleta) reduciendo la longitud de algunas palabras de código usadas en el procedimiento de binarización del parámetro mientras se mantiene una capacidad de descodificación única. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para construir un código binario truncado usando un valor máximo de un parámetro dado (por ejemplo, tal como el número de nuevas entradas de paleta). Las técnicas de ejemplo para la codificación binaria truncada se describen en http://en.wikipedia.org/wiki/Truncated_binary_encoding.

[118] De forma similar, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar códigos truncados unarios o exponenciales Golomb o Golomb-Rice para señalar y descodificar el número de nuevas entradas de paleta, basándose en un valor máximo posible del número de nuevas entradas de paleta. Por ejemplo, si (MAX_PLT_SIZE - numPredPalette) = 3, entonces el codificador de vídeo 20 puede usar un código unario truncado para señalar tres como 000 en vez de 0001 (por ejemplo, como se señalaría cuando se usa el código unario regular). En el caso de los códigos Golomb o Golomb-Rice exponenciales truncados, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para reducir en uno la longitud del prefijo para el intervalo que contiene el valor máximo. Por lo tanto, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para cambiar el prefijo de 000...001 a 000 ... 000. De forma similar, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para reducir el número de bits de sufijo en el procedimiento de binarización para ese intervalo dependiendo del valor máximo.

[119] Para bloques grandes (y/o CU grandes), el tamaño de la paleta tiende a ser el tamaño de paleta máximo. Por lo tanto, en algunos casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para asignar la binarización de (MAX_PLT_SIZE - numPredPalette) de forma inversa a la habitual, es decir, con las longitudes de palabra de código más cortas asignadas a los valores más grandes de (MAX_PLT_SIZE - numPredPalette), y las longitudes de palabra de código más largas asignadas a los valores más pequeños de (MAX_PLT_SIZE - numPredPalette). En algunos ejemplos, en lugar de usar 0's seguido por un 1 para señalar códigos unarios/unarios truncados, o como un prefijo de familia de códigos Golomb-Rice o Golomb exponencial o Golomb-Rice y Golomb exponencial concatenados, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar 1 seguido de 0.

[120] Además, otras variaciones son posibles. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para interpretar el primer bit en códigos tales como un indicador para indicar si el número de entradas nuevas es cero o distinto de cero. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para interpretar los bits restantes como un número de entradas de paleta nuevas menos 1. En un ejemplo con fines ilustrativos, un valor máximo de nuevas entradas de paleta puede ser ocho y el número de nuevas entradas de paleta puede ser tres. Usando el código unario truncado, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para determinar que la binarización sea 0001. Si el codificador de vídeo 20 y el descodificador de vídeo 30 están configurados para interpretar el primer bit como un indicador (por ejemplo, 0: una o más entradas nuevas de paleta, 1: cero nuevas entradas de paleta), el resto de los bits (001) indican que hay dos nuevas entradas de paleta. Cuando se usan códigos truncados, el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden configurar para ajustar el valor máximo hacia abajo en uno.

[121] En otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para interpretar el indicador descrito anteriormente en sentido inverso. En este caso, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para interpretar un valor de indicador de 1 como una o más entradas nuevas de paleta y un valor de indicador de 0 como cero nuevas entradas de paleta. En tal caso, los bits para señalar tres nuevas entradas de paleta con un valor máximo de ocho son 1001.

[122] En otros ejemplos, el concepto del indicador descrito anteriormente puede extenderse a otros códigos tales como Golomb exponencial, Golomb-Rice o similares. Por ejemplo, cuando el valor máximo para nuevas entradas de paleta es distinto de cero, el codificador de vídeo 20 puede configurarse para señalar un indicador que indica si hay nuevas entradas distintas de cero. Si el indicador indica que hay entradas nuevas distintas de cero, el número de nuevas entradas menos una puede señalarse usando Golomb exponencial, Golomb-Rice, concatenación de Golomb-Rice exponencial y Golomb-Rice o códigos similares o sus versiones truncadas. Cuando se utilizan versiones truncadas, el valor máximo se puede ajustar hacia abajo en uno.

[123] En algunos ejemplos, el indicador puede codificarse mediante contexto usando CABAC, mientras que el resto de bins (por ejemplo, para nuevas entradas de paleta menos 1) pueden codificarse mediante obtención. De forma alternativa, tanto el indicador como el resto de bins (para nuevas entradas de paleta menos 1) pueden codificarse como obtención. En algunos casos, un número fijo de bins de prefijos del código para las nuevas entradas de paleta

menos 1 puede codificarse mediante contexto usando CABAC y el resto de los bins pueden codificarse mediante obtención.

[124] De acuerdo con los aspectos de esta divulgación, como se indicó anteriormente, el elemento sintáctico MAX_PLT_SIZE puede señalarse en un conjunto de parámetros, tal como un SPS. En otros ejemplos, el elemento sintáctico MAX_PLT_SIZE se puede señalar en un VPS, conjunto de parámetros de imagen (PPS), cabecera de fragmento, a nivel de bloque (por ejemplo, con sintaxis señalizada para una LCU o CU), o en otro lugar. En algunos ejemplos, de acuerdo con aspectos de esta divulgación, se pueden especificar diferentes tamaños de paleta máximos para diferentes tamaños de bloques. En otros ejemplos, el tamaño de paleta máximo puede depender de un perfil o la profundidad de bits de los datos de vídeo que se codifican. Por ejemplo, para una profundidad de bits de entrada (o profundidad de bits de perfil) más grande, el elemento sintáctico MAX_PLT_SIZE se puede usar para especificar un tamaño de paleta máximo relativamente mayor. En otros ejemplos más, el tamaño de paleta máximo puede depender de forma adicional o alternativa del formato cromático de los datos de vídeo que se codifican. Por ejemplo, el elemento sintáctico MAX_PLT_SIZE puede usarse para especificar un tamaño de paleta máximo relativamente más pequeño para entradas monocromáticas que para formatos de submuestreo de croma 4:2:0, que a su vez pueden tener tamaños más pequeños que las entradas formateadas de submuestreo de croma 4:4:4.

[125] De acuerdo con los aspectos de esta divulgación, en lugar de señalar el elemento sintáctico MAX_PLT_SIZE de la manera descrita anteriormente, los datos que indican (MAX_PLT_SIZE-1) pueden señalarse, porque un elemento sintáctico MAX_PLT_SIZE que es igual a cero puede ser inválido debido a la deshabilitación de la paleta completamente.

[126] En otro ejemplo, en lugar de señalar un indicador independiente en el nivel de VPS, SPS, PPS o cabecera de fragmento para habilitar/deshabilitar el modo de paleta, el codificador de vídeo 20 puede configurarse para señalar solo el elemento sintáctico MAX_PLT_SIZE. En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para interpretar un elemento sintáctico MAX_PLT_SIZE de 0 como modo de paleta de deshabilitación. Es decir, al recibir un elemento sintáctico de tamaño de paleta (por ejemplo, el elemento sintáctico MAX_PLT_SIZE), el descodificador de vídeo 30 puede determinar que el modo de paleta se ha deshabilitado basándose en el elemento sintáctico. El elemento sintáctico MAX_PLT_SIZE o (MAX_PLT_SIZE - 1) se puede señalar utilizando códigos de longitud fija (suponiendo un límite normativo en MAX_PLT_SIZE) o códigos Golomb-Rice o Golomb exponencial.

[127] Como se indicó anteriormente, las técnicas de esta divulgación también incluyen datos de codificación que indican un tamaño de predictor de paleta máximo. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un elemento sintáctico MAX_PLT_PREDICTOR_SIZE en el VPS, SPS, PPS, cabecera de fragmento, a nivel de bloque o en otro lugar que indique un tamaño de predictor de paleta máximo. En algunos ejemplos, en lugar de señalar el elemento sintáctico MAX_PLT_PREDICTOR_SIZE, se puede señalar (MAX_PLT_PREDICTOR_SIZE-1). En otros ejemplos más, el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden configurar para codificar otros datos que indiquen un tamaño de predictor de paleta máximo.

[128] En los ejemplos particulares descritos en el presente documento, el elemento sintáctico MAX_PLT_PREDICTOR_SIZE o (MAX_PLT_PREDICTOR_SIZE - 1) puede señalarse utilizando códigos de longitud fija (por ejemplo, suponiendo un límite normativo en MAX_PLT_PREDICTOR_SIZE) o códigos Golomb Rice o Golomb exponencial. En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para suponer (por ejemplo, determinar automáticamente) que el tamaño indicado por el elemento sintáctico MAX_PLT_PREDICTOR_SIZE es mayor o igual que un tamaño de paleta máximo (por ejemplo, como lo indica mediante un elemento sintáctico MAX_PLT_SIZE). En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar (MAX_PLT_PREDICTOR_SIZE - MAX_PLT_SIZE) utilizando códigos de longitud fija o códigos Golomb-Rice o Golomb exponencial. Por consiguiente, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican una delta (por ejemplo, diferencia) entre el tamaño de predictor de paleta máximo y el tamaño de paleta máximo.

[129] En ejemplos en los que el tamaño de paleta máximo y el tamaño de predictor de paleta máximo se señalan a nivel de SPS, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican el número de entradas nuevas usando un código unario truncado. El número de entradas nuevas más el número de entradas predichas del predictor de paleta juntas no puede exceder el tamaño de paleta máximo señalado en el SPS. Sin embargo, si el tamaño de paleta máximo señalado en el SPS es relativamente grande, el número de entradas nuevas puede ser superior a 31. En este caso, el código unario truncado excede una longitud de 32 bits, lo cual puede no ser deseable para las implementaciones de software y hardware.

[130] Para abordar esto, de acuerdo con los aspectos de esta divulgación, en un ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para restringir el número de nuevas entradas de modo que la longitud del código para señalar el número de nuevas entradas no sea superior a 32. Por ejemplo, si se usa un código unario o unario truncado para señalar el número de entradas nuevas, el número de entradas nuevas puede restringirse

a 31. Debe entenderse que una restricción de longitud de 32 es simplemente un ejemplo (por ejemplo, pueden usarse de forma alternativa otras restricciones de longitud).

[131] En el borrador 2 de memoria descriptiva textual de las extensiones de codificación de contenido de la pantalla HEVC, (Rajan Joshi et al., "High Efficiency Video Coding (HEVC) Screen Content Coding: Draft 2" [Codificación de contenido de la pantalla de codificación de vídeo de alta eficiencia (HEVC): Borrador 2], JCTVC-S1005, Sapporo, JP, 30 de junio - 9 de julio de 2014 (en adelante JCTVC-S1005), el código unario truncado se utiliza para señalar el número de nuevas entradas de paleta con el valor máximo igual al tamaño de paleta máximo señalado en el SPS (`palette_max_size`) menos el número de entradas de paleta que se predicen desde el predictor de paleta. Usando la restricción propuesta, el valor máximo puede modificarse para que sea el menor de 32 y la diferencia entre el tamaño de paleta máximo señalado en el SPS (`palette_max_size`) y el número de entradas de paleta predichas desde el predictor de paleta. Si se realiza tal modificación al valor máximo y se utiliza la codificación unaria truncada de JCTVC-S1005, el número máximo de nuevas entradas de paleta puede ser 32 (en lugar de 31) sin que la longitud del código sea superior a 32 bits.

[132] En algunos ejemplos, en lugar de codificación unaria truncada, si el codificador de vídeo 20 y el decodificador de vídeo 30 están configurados usan otro código tal como Golomb exponencial o su versión truncada, las nuevas entradas de paleta máximas permitidas pueden modificarse apropiadamente para que la longitud no sea superior a 32. Si hay una serie de valores que tienen una longitud de palabra de código de 32, el codificador de vídeo 20 y el decodificador de vídeo 30 se pueden configurar para elegir el valor más alto posible para el valor máximo permitido para el número de nuevas entradas de paleta.

[133] La restricción descrita en el presente documento sobre el número máximo de nuevas entradas de paleta se puede convertir en una restricción normativa. Por ejemplo, el codificador de vídeo 20 se puede configurar para generar un flujo de bits con la restricción y el decodificador de vídeo 30 se puede configurar para confiar en la restricción en un flujo de bits conforme.

[134] De acuerdo con aspectos de esta divulgación, en un ejemplo, la semántica del elemento sintáctico de entradas señaladas `palette_num` puede cambiarse con respecto a JCTVC-S1005 de la siguiente manera: el elemento sintáctico `palette_num_signaled_entries` especifica el número de entradas en la paleta actual que están señaladas explícitamente. El valor del elemento sintáctico `palette_num_signaled_entries` estará en la gama entre 0 y 31, inclusive. Cuando no está presente el elemento sintáctico `palette_num_signaled_entries`, se infiere que es igual a 0.

[135] Además, el valor de la variable `CurrentPaletteSize` especifica el tamaño de la paleta actual y se obtiene de la siguiente manera:

Si `palette_share_flag[x0][y0]` es igual a 1,

`CurrentPaletteSize` = `PreviousPaletteSize` (7-71)

De lo contrario (`palette_share_flag[x0][y0]` es igual a 0)

`CurrentPaletteSize` = `paletteNumPredictedEntries` +

`palette_num_signaled_entries`(7-72)

[136] En el ejemplo anterior, el valor de `CurrentPaletteSize` puede estar en el intervalo de 0 a `palette_max_size`, inclusive.

[137] De acuerdo con aspectos de esta divulgación, si el valor máximo se modifica de la manera descrita anteriormente, el valor del elemento sintáctico `palette_num_signaled_entries` se puede modificar de manera que el valor esté en el intervalo de 0 a 32, inclusive.

[138] En otro ejemplo, el tamaño de paleta máximo puede señalarse en el SPS y puede limitarse a 31. La limitación del tamaño se puede lograr aplicando un límite superior en el elemento sintáctico `palette_max_size` en la semántica del elemento sintáctico `palette_max_size`, de modo que el elemento sintáctico `palette_max_size` especifique el tamaño máximo permitido de la paleta. El valor del elemento sintáctico `palette_max_size` estará en la gama entre 0 y 31, inclusive. Cuando no está presente, el valor del elemento sintáctico `palette_max_size` se infiere que es 0. En algunos ejemplos, en lugar de 31, el valor puede restringirse a 32.

[139] En otro ejemplo, el valor máximo del elemento sintáctico `palette_max_size` puede restringirse de modo que si el número de nuevas entradas de paleta es igual a `palette_max_size`, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para codificar el número de nuevas entradas de paleta usando un código que no exceda los 32 bits. En otro ejemplo más, el valor máximo de número de nuevas entradas de paleta siempre puede

estar limitado a 31, independientemente del código utilizado para codificar el valor máximo. En otro ejemplo más, el valor máximo puede estar limitado a 32.

[140] Las técnicas de esta divulgación también se refieren a la señalización de salida a nivel de bloque (por ejemplo, para una CU o LCU). Por ejemplo, de acuerdo con los aspectos de esta divulgación, uno o más elementos sintácticos pueden indicar, a nivel de bloque (por ejemplo, un nivel CU), si alguna de las muestras del bloque está codificada como una muestra de salida (por ejemplo, una muestra que no tiene un valor de color representado en una paleta para codificar el bloque). Como se indicó anteriormente, el uno o más elementos sintácticos pueden denominarse sintaxis de salida a nivel de bloque. Nuevamente, la sintaxis a nivel de bloque puede referirse a la sintaxis codificada o determinada con un bloque de datos de vídeo, como una C1.1 o LCU, y no a la sintaxis que puede incluirse en una cabecera de fragmento o con píxeles individuales de datos de vídeo.

[141] En casos en los que al menos una muestra en un bloque de muestras codificadas usando codificación de paleta se codifica como una muestra de salida, las técnicas de esta divulgación se pueden usar para señalar la existencia de dicho modo. En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un indicador (que puede denominarse indicador de salida a nivel de bloque) que indica si alguna de las muestras del bloque codificado está codificada como una muestra de salida. En algunos casos, un valor de indicador de cero puede indicar que ninguna de las muestras o píxeles del bloque se codifica como muestras de salida. Un valor de indicador de uno puede indicar que al menos una muestra o píxel del bloque se codifica como una muestra de salida. Por lo tanto, la sintaxis de salida a nivel de bloque puede indicar, para todas las muestras de un bloque de datos de vídeo, si se codifica al menos una muestra del bloque sin usar un índice para unos valores de paleta de colores para el bloque, por ejemplo, se codifica usando el modo de salida.

[142] De acuerdo con aspectos de esta divulgación, la sintaxis descrita anteriormente puede lograr un ahorro de bits con respecto a técnicas de muestras de salida de señalización sin una indicación a nivel de bloque. Por ejemplo, en casos en que la sintaxis indica que no hay muestras de un bloque codificadas como muestras de salida (por ejemplo, el indicador descrito anteriormente es cero), el codificador de vídeo 20 y el descodificador de vídeo 30 pueden no codificar ninguna otra sintaxis asociada con muestras de salida para el bloque. Por ejemplo, con respecto a la señalización de salida explícita descrita en el presente documento, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden omitir la codificación de los indicadores del modo de salida a nivel de muestra. Con respecto a la señalización de salida implícita, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden omitir la codificación del índice adicional para la paleta que indica la muestra de salida. En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 solo pueden codificar un indicador de SPoint para distinguir entre el modo CopyFromTop y el modo de valor.

[143] En algunos ejemplos, el indicador descrito anteriormente puede señalarse antes de las entradas de paleta para un bloque o CU que se está codificando actualmente. En otros ejemplos, el indicador descrito anteriormente puede señalarse después de las entradas de la paleta para un bloque o CU que se está codificando actualmente. En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar el contexto del indicador descrito anteriormente. En dichos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar los contextos basados en el tamaño de bloque o CU y/o el tamaño de paleta para el bloque o CU actual.

[144] En algunos casos, el uso de muestras de salida puede variar según el tamaño del bloque. Por ejemplo, el uso de muestras de salida puede ser menos frecuente en bloques relativamente pequeños. En tales casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para determinar que un bloque no incluye ninguna muestra de salida y omitir la codificación del indicador descrito anteriormente, logrando así un ahorro de bits. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden no codificar el indicador a nivel de bloque para bloques de 8x8, donde es mucho menos probable que se usen muestras de salida que en bloques de mayor tamaño. De manera similar, para tamaños de bloques grandes (por ejemplo, bloques de 64x64 píxeles o mayores), el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden configurar para determinar que siempre haya muestras codificadas como muestras de salida. En tales casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que el indicador de salida a nivel de bloque para un bloque es igual a uno (por ejemplo, al menos una muestra es una muestra de salida) y omitir la codificación del indicador de salida a nivel de bloque (por ejemplo, una indicación del indicador de salida a nivel de bloque no está incluida en el flujo de bits).

[145] Las técnicas de esta divulgación también se refieren a la codificación de un bloque de muestras basándose en si las muestras de un bloque están codificadas como muestras de salida. Por ejemplo, como se indicó anteriormente, las técnicas de esta divulgación pueden usarse para indicar si alguna muestra está codificada como muestra de salida de un bloque codificado con paleta. En casos en los que un bloque no incluye muestras de salida y cuando el tamaño de una paleta es uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para determinar automáticamente que todas las muestras del bloque tienen el mismo valor de índice (por ejemplo, la única entrada de la paleta). El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden, por lo tanto, omitir la codificación de todos los otros datos usados para determinar los índices de paleta del bloque. Por ejemplo, el

codificador de vídeo 20 y el descodificador de vídeo 30 pueden omitir la codificación del indicador de SPoint, la señalización de índice y los datos asociados con las series de índices de paleta.

[146] En un ejemplo con fines ilustrativos, el descodificador de vídeo 30 puede descodificar un indicador de salida a nivel de bloque que indica que no hay muestras en el bloque actual que estén codificadas como muestras de salida (por ejemplo, el indicador es igual a cero). El descodificador de vídeo 30 también puede descodificar datos que indican que la paleta del bloque tiene una sola entrada (por ejemplo, datos que indican que el tamaño de la paleta es uno) o descodificar una paleta que tiene una sola entrada. En este ejemplo, basándose en que ambas condiciones se evalúan como verdaderas (por ejemplo, ninguna muestra es muestra de salida y el tamaño de la paleta es uno), el descodificador de vídeo 30 puede determinar automáticamente que todos los índices de paleta del bloque son iguales a la entrada única incluida en la paleta. El descodificador de vídeo 30 también puede omitir la descodificación de otros datos usados para determinar los índices de paleta del bloque (por ejemplo, tales como indicadores SPoint, índices de paleta e información de serie).

[147] En otro ejemplo, de acuerdo con aspectos de esta divulgación, cuando el tamaño de una paleta es uno, las muestras de salida pueden terminar las series de muestras codificadas con el índice de paleta cero. Es decir, una serie de índices de paleta puede verse interrumpida por una posición codificada como una muestra de salida. En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para omitir la codificación del indicador de SPoint. Además, en este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para inferir que el modo para los índices de paleta es el modo de valor así como el índice del modo de valor (por ejemplo, con solo una entrada en la paleta, no será necesario indicar el índice para el modo de valor). En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir que la muestra que sigue inmediatamente a una serie se codifica como una muestra de salida y omite la codificación de la sintaxis relacionada con la salida.

[148] Las técnicas de esta divulgación también se refieren a datos de codificación que indican un valor de serie de una serie de índices de paleta en la codificación de paleta. Por ejemplo, como se indicó anteriormente, un valor de serie puede indicar un número de muestras consecutivas (por ejemplo, una serie de muestras) en un orden de escaneado particular en un bloque codificado con paleta que están codificadas juntas. En algunos casos, la serie de muestras también se puede denominar una serie de índices de paleta, porque cada muestra de la serie tiene un índice asociado a una paleta.

[149] Un valor de serie puede indicar una serie de índices de paleta que están codificados usando el mismo modo de codificación de paleta. Por ejemplo, con respecto al modo de valor, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar un valor de índice y un valor de serie que indica un número de muestras consecutivas en un orden de escaneado que tienen el mismo valor de índice y que están siendo codificadas con el valor de índice. Con respecto al modo CopyFromTop, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar una indicación de que un índice para el valor de muestra actual se copia basándose en un índice de una muestra contigua anterior (por ejemplo, una muestra que está posicionada sobre la muestra que está siendo codificada en un bloque) y un valor de serie que indica un número de muestras consecutivas en un orden de escaneado que también copia un valor de índice de una muestra contigua superior y que se está codificando con el valor del índice.

[150] Por ejemplo, de acuerdo con aspectos de esta divulgación, los datos que indican una serie de índices de paleta en un bloque de datos de vídeo pueden codificarse basándose en un valor de serie máximo posible para el bloque. Es decir, el codificador 20 y el descodificador 30 pueden determinar, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta, una longitud de serie de una serie de índices de paleta codificados con el índice de paleta del píxel. El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden determinar una longitud de serie máxima para una serie máxima de índices de paleta que pueden codificarse con el índice de paleta del píxel. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden entonces codificar datos que indican la longitud de serie basándose en la longitud de serie máxima determinada.

[151] En un ejemplo con fines ilustrativos, el número total de muestras en un bloque de datos de vídeo puede ser N y cada una de las muestras puede estar indexada de 0 a (N-1). Para la muestra con la posición j, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar el máximo valor de serie posible como (N-j-1). Se debe tener en cuenta que el valor de serie indica el número de muestras posteriores que se codifican con el mismo modo de codificación de paleta (por ejemplo, el modo de valor o el modo CopyFromTop) como la muestra actual. De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican el valor de serie usando una binarización truncada, teniendo en cuenta el máximo valor de serie posible. En general, la binarización truncada puede incluir cualquier técnica que utilice información sobre un valor máximo posible de un parámetro particular que se señala (por ejemplo, como el número de nuevas entradas de paleta) reduciendo la longitud de algunas palabras de código usadas en el procedimiento de binarización del parámetro mientras se mantiene una capacidad de descodificación única. Por ejemplo, se puede usar un código binario truncado basándose en el valor máximo posible de una serie. De manera similar, los códigos truncados unarios o exponenciales Golomb o Golomb Rice se pueden usar para codificar y descodificar el valor de serie, basándose en el valor máximo posible de una serie. En algunos ejemplos, la binarización truncada puede ser una combinación de códigos exponenciales Golomb y Golomb-Rice.

[152] Por ejemplo, una palabra de código Exp-Golomb (EGk) de orden $k^{\text{ésimo}}$ se compone de dos partes, un prefijo y un sufijo. Para un entero x sin signo dado, la parte de prefijo de la palabra de código EGk consiste en un código unario que corresponde al valor de:

$$l(x) = \left\lfloor \log_2 \left(\frac{x}{2^k} + 1 \right) \right\rfloor$$

La parte del sufijo se calcula como la representación binaria de $x - 2^k(2^{l(x)} - 1)$ utilizando $k + l(x)$ bits.

[153] Como ejemplo, la tabla 1 siguiente incluye varias palabras de código para EG0.

TABLA 1 - Ejemplo de EG0

Valor x	Palabra de código (prefijo -sufijo)	Longitud de palabra de código
0	1	1
1	01-0	3
2	01-1	3
3	001-00	5
4	001-01	5
5	001-10	5
6	001-11	5

[154] En la Solicitud Provisional de Estados Unidos n.º 62/019,223, presentada el 20 de junio de 2014, un valor de serie se codifica utilizando el código Exp-Golomb de 2.º orden.

[155] De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican un valor de serie usando un código Exp-Golomb truncado. Por ejemplo, una palabra de código Exp-Golomb (TEGk) truncado de k -ésimo orden también está compuesta de dos partes, un prefijo y un sufijo. El prefijo puede ser un prefijo unario y el sufijo puede ser un sufijo binario. Por ejemplo, para un entero x sin signo dado y su valor de serie más grande posible X_{max} (por ejemplo, la longitud de serie máxima), la parte de prefijo de la palabra de código EGk consiste en un código unario truncado que corresponde al valor de:

$$l(x) = \left\lfloor \log_2 \left(\frac{x}{2^k} + 1 \right) \right\rfloor$$

Específicamente, el "final" del código unario se puede evitar si:

$$\left\lfloor \log_2 \left(\frac{X_{\text{max}}}{2^k} + 1 \right) \right\rfloor = l(x).$$

[156] Si el prefijo está truncado, es decir, $\left\lfloor \log_2 \left(\frac{x}{2^k} + 1 \right) \right\rfloor = l(x)$, la parte del sufijo de TEGk se calcula como la representación binaria truncada de $x - 2^k(2^{l(x)} - 1)$ utilizando $k + l(x)$ o $k + l(x) - 1$ bits. El valor máximo del símbolo para la entrada del código binario truncado es $X_{\text{max}} - 2^k(2^{l(x)} - 1)$.

[157] Si el prefijo no está truncado, la parte del sufijo de TEGk es la misma que EGk, es decir, la representación binaria de $x - 2^k(2^{l(x)} - 1)$ utilizando $k + l(x)$ bits. Como ejemplo, la tabla 1 siguiente incluye varias palabras de código para TEG0.

TABLA 2 - Ejemplos de TEG0 (X=5)

Valor x	Palabra de código (prefijo-sufijo)	Longitud de palabra de código
0	1	1
1	01-0	3

Valor x	Palabra de código (prefijo-sufijo)	Longitud de palabra de código
2	01-1	3
3	00-0	3
4	00-01	4
5	00-10	4

Si bien el ejemplo de la tabla 2 anterior ilustra ese prefijo como un número de ceros seguido de uno posterior (por ejemplo, 00... 1_), debe entenderse que en otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar un número de unos seguidos por un cero posterior (por ejemplo, 11.... 0).

[158] De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar un valor de serie usando el código TEGk descrito anteriormente. En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden, para una posición de píxel actual en un bloque (o CU), determinar el valor de serie máximo X_{max} basado en la ecuación (X_{max} = número de píxeles en la CU actual - posición actual en orden de escaneado - 1).

[159] En otro ejemplo, si el valor de serie se codifica en primer lugar usando un prefijo unario truncado, el codificador de vídeo 20 y el descodificador de vídeo 30 se pueden configurar para ajustar el valor de serie máximo en consecuencia, por ejemplo, basándose en el valor truncado. Por ejemplo, en algunos casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un valor de serie como una serie de tres indicadores: mayor que cero, mayor que uno y mayor que dos. En este ejemplo, si la serie señalada es mayor que dos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar el valor restante (por ejemplo, valor de serie - 3), potencialmente con otro procedimiento de binarización como una combinación de códigos Golomb exponencial y Golomb-Rice o el código TEGk descrito anteriormente.

[160] Sin embargo, si (N-j-1) es igual a 0, el codificador de vídeo 20 y el descodificador de vídeo 30 no codifican una serie. Del mismo modo, si (N-j-1) es igual a uno, el codificador de vídeo 20 y el descodificador de vídeo 30 solo pueden codificar el indicador de mayor que cero. Del mismo modo, si (N-j-1) es igual a dos, el codificador de vídeo 20 y el descodificador de vídeo 30 solo pueden codificar el indicador mayor que cero y el mayor de uno. Del mismo modo, si (N-j-1) es igual a tres, el codificador de vídeo 20 y el descodificador de vídeo 30 solo pueden codificar el indicador mayor que cero, el indicador mayor que uno y el indicador mayor que dos. Si (N-j-1) es mayor que tres, además del indicador mayor que cero, el indicador mayor que uno y el indicador mayor que dos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar el valor restante hasta un valor máximo de (N-j-4). De manera similar, el proceso descrito puede extenderse para usar un número de indicadores distintos de los tres, por ejemplo, indicadores que indican un valor señalizado mayor que el número M, donde M puede ser un valor no negativo comenzando desde cero.

[161] De acuerdo con los aspectos de esta divulgación, en el ejemplo anterior, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar la longitud de serie restante usando el código TEGk descrito anteriormente, con un valor de serie máximo igual a (número de píxeles en la CU actual - posición actual en orden de escaneado - 4). En otro ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un indicador que indica si el valor de serie es mayor que cero y el valor restante como la longitud de serie menos uno. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar el indicador de mayor que cero. El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden codificar datos que indican la longitud de serie menos uno usando el código TEGk con un valor máximo para el conjunto de códigos TEGk igual a la longitud de serie máxima menos uno. En un ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden establecer k igual a cero, de manera que el código TEGk sea un código TEG0.

[162] En otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar cualquier orden del código TEG descrito anteriormente para codificar elementos sintácticos para la codificación de paleta. En un ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden establecer k igual a dos, de manera que el código TEGk sea un código TEG2.

[163] Mientras que los ejemplos anteriores se describen con respecto a la codificación de un valor de serie en codificación de paleta, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para usar los códigos (tales como el código TEGk) para codificar otra sintaxis para la codificación de paleta. Por ejemplo, como se describe en mayor detalle a continuación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar los códigos descritos anteriormente para codificar un vector de predicción de paleta binaria, una longitud de serie de CopyAbove u otros valores.

[164] En Joshi et al., "Non-SCCE3: Contexts for coding index runs" [No SCCE3: Contextos para la codificación de series de índice] Equipo colaborativo conjunto en codificación de vídeo (JCT-VC) de ITU-T SG 16 WP 3 e ISO/IEC

JTC 1/SC 29/WG 11, 18.^a reunión, Sapporo, JP, 30 junio - 9 julio de 2014 , JCTVC-R0174 (en adelante, JCTVC-R0174), los autores propusieron que los contextos de las palabras clave de longitud de serie dependieran del índice si se usa el modo CopyLeft (por ejemplo, que puede funcionar de manera similar al modo CopyFromTop). Sin embargo, de acuerdo con un ejemplo de esta divulgación, si el modo de serie actual es CopyFromAbove, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar contextos para CABAC que codifican la serie basándose en el valor de índice del píxel que está posicionado sobre el píxel que está siendo actualmente codificado. En este ejemplo, el píxel contiguo superior está fuera de la CU actual, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar que el índice correspondiente es igual a una constante redefinida k . En algunos ejemplos, la constante k puede ser igual a cero.

[165] En algunos ejemplos, si el modo de paleta para codificar un píxel actual es el modo CopyFromAbove, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden seleccionar uno de dos contextos de CABAC candidatos para codificar el primer bin de la palabra de código de longitud de serie basándose en si el píxel contiguo superior tiene un índice que es igual a cero. Como otro ejemplo, si el modo de paleta para codificar un píxel actual es el modo de copia previa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden seleccionar uno de los cuatro contextos de CABAC candidatos para codificar el primer bin de la palabra de código de longitud de serie basándose en si el índice es igual a cero, uno, dos o más grande que dos.

[166] La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de codificador de vídeo 20 que puede implementar las técnicas de esta divulgación. La FIG. 2 se proporciona con fines explicativos y no debe considerarse limitante de las técnicas ampliamente ejemplificadas y descritas en esta divulgación. Con fines explicativos, 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.

[167] El codificador de vídeo 20 representa un ejemplo de un dispositivo que puede configurarse para realizar técnicas para la codificación de vídeo basada en paleta de acuerdo con 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 CU o PU en codificación HEVC, usando codificación basada en paleta o codificación no basada en paleta. Los modos de codificación no basados en paleta pueden referirse a varios modos de codificación temporal inter-predictiva o modos de codificación espacial intra-predictiva, tales como los diversos modos de codificación especificados por el borrador de HEVC 10. 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 píxel 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 con entradas en la paleta correspondientes, respectivamente, a los valores de píxel seleccionados en la paleta. La información señalada puede ser utilizada por el descodificador de vídeo 30 para descodificar datos de vídeo.

[168] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluyen una unidad de procesamiento de predicción 100, una memoria de datos de vídeo 101, una unidad de generación residual 102, una unidad de procesamiento de transformada 104, una unidad de cuantificación 106, una unidad de cuantificación inversa 108, una unidad de procesamiento de transformada inversa 110, una unidad de reconstrucción 112, una unidad de filtro 114, una memoria intermedia de imágenes descodificadas 116, y una unidad de codificación por entropía 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de inter-predicción 120 y una unidad de procesamiento de intra-predicción 126. La unidad de procesamiento 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 paleta 122 configurada para realizar diversos aspectos de las técnicas de codificación basadas en paleta descritas en esta divulgación. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[169] La memoria de datos de vídeo 101 puede almacenar datos de vídeo para 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, de la fuente de vídeo 18. La memoria intermedia de imágenes descodificadas 116 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para usar en la codificación de datos de vídeo mediante el codificador de vídeo 20, por ejemplo, en modos de intra-codificación o inter-codificación. La memoria de datos de vídeo 101 y la memoria intermedia de imágenes descodificadas 116 pueden estar formadas por cualquiera de una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magneto-resistiva (MRAM), RAM resistiva (RRAM), u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 101 y la memoria intermedia de imágenes descodificadas 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 chip con otros componentes del codificador de vídeo 20, o fuera de chip con respecto a esos componentes.

[170] 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 con bloques de árbol de codificación (CTB) luma de igual tamaño y CTB correspondientes de la imagen. Como parte de la codificación de una CTU, la unidad de procesamiento de predicción 100 puede realizar división de árbol cuádruple para dividir los

CTB de la CTU en bloques progresivamente más pequeños. El bloque más pequeño puede ser bloques de codificación de CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un CTB asociado con 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.

[171] El codificador de vídeo 20 puede codificar CU de una CTU para generar representaciones codificadas de las CU (es decir, CU codificadas). Como parte de la codificación de una CU, la unidad de procesamiento de predicción 100 puede dividir los bloques de codificación asociados con la CU entre una o más PU de la CU. Por lo tanto, cada PU puede asociarse con un bloque de predicción de luma y bloques de predicción de croma correspondientes. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden soportar 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 soportar tamaños de PU de $2N \times 2N$ o $N \times N$ para intra-predicción, y tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ o similares para inter-predicción. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden soportar división asimétrica para tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$ para inter-predicción.

[172] La unidad de procesamiento 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 dependiendo de si la PU está en un fragmento I, un fragmento P o un fragmento B. En un fragmento en 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 el modo I, el bloque predicho se forma usando predicción espacial a partir de bloques contiguos previamente codificados dentro de la misma trama.

[173] 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, "RefPicList0") 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 puede generar un índice de referencia que indique una posición en RefPicList0 de la imagen de referencia que contiene la región de referencia para la PU. Además, la unidad de estimación de movimiento puede generar un MV que indica un desplazamiento espacial entre un bloque de codificación de la PU y una ubicación de referencia asociada con la región de referencia. Por ejemplo, el MV puede ser un vector bidimensional que proporciona una desviación de las coordenadas en la imagen descodificada actual a las coordenadas de una imagen de referencia. La unidad de estimación de movimiento puede emitir el índice de referencia y el MV 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.

[174] 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 predicción unidireccional para la PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia de RefPicList0 o una segunda lista de imágenes de referencia ("RefPicList1") 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 RefPicList0 o RefPicList1 de la imagen de referencia que contiene la región de referencia, un MV que indica un desplazamiento espacial entre un bloque de predicción de la PU y una ubicación de referencia asociada con la región de referencia, y uno o más indicadores de dirección de predicción que indican si la imagen de referencia está en RefPicList0 o RefPicList1. 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.

[175] Para realizar inter-predicción bidireccional para una PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia en RefPicList0 para una región de referencia para la PU y también puede buscar las imágenes de referencia en RefPicList1 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 RefPicList0 y RefPicList1 de las imágenes de referencia que contienen las regiones de referencia. Además, la unidad de estimación de movimiento puede generar MV que indican desplazamientos espaciales entre la ubicación de referencia asociada con 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.

[176] De acuerdo con diversos ejemplos de esta divulgación, el codificador de vídeo 20 puede configurarse para realizar una codificación basada en paleta. Con respecto al marco de HEVC, como ejemplo, las técnicas de

codificación basadas en paleta se pueden configurar para usarse como modo de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basadas en paleta pueden configurarse para usarse como un modo PU en el marco de HEVC. De acuerdo con esto, todos los procesos divulgados descritos en el presente documento (a lo largo de esta divulgación) en el contexto de un modo CU pueden, de forma adicional o alternativa, aplicarse a la PU. Sin embargo, estos ejemplos basados en HEVC no se deben considerar una restricción o limitación de las técnicas de codificación basadas en paleta descritas en el presente documento, ya que tales técnicas se pueden aplicar para funcionar independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación de paleta pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[177] La unidad de codificación basada en paleta 122, por ejemplo, puede realizar una descodificación basada en paleta cuando se selecciona un modo 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 píxel de al menos algunas posiciones de un bloque de datos de vídeo, y señalar información asociando al menos algunas de las posiciones del bloque de datos de vídeo con entradas en la paleta correspondientes, respectivamente, a los valores de píxel seleccionados. Aunque se describen diversas funciones realizadas por la unidad de codificación basada en paleta 122, algunas o todas estas funciones pueden ser realizadas por otras unidades de procesamiento, o una combinación de diferentes unidades de procesamiento.

[178] De acuerdo con aspectos de esta divulgación, la unidad de codificación basada en paleta 122 puede configurarse para realizar cualquier combinación de las técnicas para codificación de paletas descritas en el presente documento. Por ejemplo, de acuerdo con aspectos de esta divulgación, la unidad de codificación basada en paleta 122 puede determinar un valor de un elemento sintáctico que indica, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque está codificada con un primer modo de paleta, donde el primer modo de paleta incluye codificar la muestra respectiva del bloque sin usar un índice a unos valores de paleta de colores para el bloque. Por ejemplo, la unidad de codificación basada en paleta 122 puede determinar un valor de un elemento sintáctico a nivel de bloque que indique si cualquier muestra del bloque está codificada como una muestra de salida (como una muestra que no tiene un valor de color representado en una paleta para codificando el bloque). En algunos ejemplos, la unidad de codificación basada en paleta 122 puede determinar un indicador de salida para un bloque que indica si cualquier muestra del bloque está codificada como una muestra de salida.

[179] De forma adicional o alternativa, la unidad de codificación basada en paleta 122 puede determinar al menos uno de los datos que indica un tamaño de paleta máximo de unos valores de paleta de colores para codificar un bloque de datos de vídeo o datos que indica un tamaño de predictor de paleta máximo de un predictor de paleta determinar los valores de paleta de colores. Por ejemplo, la unidad de codificación basada en paleta 122 puede incluir tales datos en un conjunto de parámetros, tal como un SPS.

[180] De forma adicional o alternativa, la unidad de codificación basada en paleta 122 puede determinar, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores para codificar el píxel, una longitud de serie de índices de paleta que se están codificando con el índice de paleta del píxel. Es decir, la unidad de codificación basada en paleta 122 que puede determinar que un valor de índice para una muestra particular está siendo codificada con una serie de otros índices de paletas posteriores. La unidad de codificación basada en paleta 122 también puede determinar una longitud de serie máxima para una serie máxima de índices de paleta capaces de ser codificados con el índice de paleta del píxel. Como se describe en mayor detalle a continuación con respecto a la unidad de codificación por entropía 118, el codificador de vídeo 20 puede codificar datos que indican la longitud de serie basándose en la longitud de serie máxima determinada.

[181] 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 PU en fragmentos I, fragmentos P y fragmentos B.

[182] Para realizar la intra-predicción en una PU, la unidad de procesamiento de intra-predicción 126 puede usar múltiples modos 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 PU contiguas 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 arriba a abajo, de izquierda a derecha para las PU, CU y CTU. La unidad de procesamiento de intra-predicción 126 puede usar varios números de modos de intra-predicción, por ejemplo, 33 modos de intra-predicción direccional. En algunos ejemplos, el número de modos de intra-predicción puede depender del tamaño de la región asociada con la PU.

[183] La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para 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 la 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.

[184] 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 bloque predictivo seleccionado correspondiente de una PU de la CU.

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

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

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

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

[189] La unidad de filtro 114 puede realizar una o más operaciones de desbloqueo para reducir los artefactos de bloqueo en los bloques de codificación asociados con una CU. La memoria intermedia de imágenes descodificadas 116 puede almacenar los bloques de codificación reconstruidos después de que la unidad de filtro 114 realice la 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 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 descodificadas 116 para realizar intra-predicción en otras PU en la misma imagen que la CU.

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

[191] De acuerdo con aspectos de esta divulgación, la unidad de codificación por entropía 118 puede configurarse para codificar datos de paleta usando un código TEGk, como se describió anteriormente con respecto al ejemplo de la FIG. 1. En particular, de acuerdo con los aspectos de esta divulgación, la unidad de codificación por entropía 118 puede codificar datos que indican una longitud de serie para una serie de índices de paleta basándose en una longitud de serie máxima determinada. En algunos ejemplos, la unidad de codificación por entropía 118 puede codificar la longitud de serie usando un código TEG2.

[192] La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de descodificador de vídeo 30 que está configurado para implementar las técnicas de esta divulgación. La FIG. 3 se proporciona con fines explicativos y no se limita a las técnicas ampliamente ejemplificadas y descritas en esta divulgación. Con fines explicativos, esta divulgación describe el descodificador 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.

[193] El codificador de vídeo 20 representa un ejemplo de un dispositivo que puede configurarse para realizar técnicas para la codificación de vídeo basada en paleta de acuerdo con varios ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 20 puede estar configurado para descodificar selectivamente varios bloques de datos de vídeo, tales como CU o PU en codificación HEVC, usando codificación basada en paleta o codificación no basada en paleta. Los modos de codificación no basados en paleta pueden referirse a varios modos de codificación temporal inter-predictiva o modos de codificación espacial intra-predictiva, tales como los diversos modos de codificación especificados por el borrador de HEVC 10. El descodificador de vídeo 30, en un ejemplo, puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxel, reciba información que asocie al menos algunas ubicaciones de píxeles en un bloque de datos de vídeo con entradas en la paleta, seleccione valores de píxel en la paleta basándose en la información, y reconstruya los valores de píxel del bloque basándose en los valores de píxel seleccionados en la paleta.

[194] En el ejemplo de la FIG. 3, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 150, una memoria de datos de vídeo 151, una unidad de procesamiento de predicción 152, una unidad de cuantificación inversa 154, una unidad de procesamiento de transformada inversa 156, una unidad de reconstrucción 158, una unidad de filtro 160 y una memoria intermedia de imágenes descodificadas 162. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 164 y una unidad de procesamiento de intra-predicción 166. El descodificador de vídeo 30 también incluye una unidad de descodificación basada en paleta 165 configurada para realizar diversos aspectos de las técnicas de codificación basadas en paleta descritas en esta divulgación. En otros ejemplos, el descodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

[195] La memoria de datos de vídeo 151 puede almacenar datos de vídeo, tales como un flujo de bits de vídeo codificado, para ser descodificados por los componentes del descodificador de vídeo 30. Los datos de vídeo almacenados en la memoria de datos de vídeo 151 pueden obtenerse, por ejemplo, del canal 16, por ejemplo, desde una fuente de vídeo local, como una cámara, mediante comunicación de datos de vídeo por cable o red inalámbrica, o accediendo a medios físicos de almacenamiento de datos. La memoria de datos de vídeo 151 puede formar una memoria intermedia de imágenes codificadas (CPB) que almacena datos de vídeo codificados a partir de un flujo de bits de vídeo codificado. La memoria intermedia de imágenes descodificadas 162 puede ser una memoria de imágenes de referencia que almacena datos de vídeo de referencia para usar en la descodificación de datos de vídeo mediante el descodificador de vídeo 30, por ejemplo, en modos de intra-codificación o inter-codificación. La memoria de datos de vídeo 151 y la memoria intermedia de imágenes descodificadas 162 pueden estar formadas por cualquiera de una variedad de dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magneto-resistiva (MRAM), RAM resistiva (RRAM), u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 151 y la memoria intermedia de imágenes descodificadas 162 pueden ser proporcionadas por el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 151 puede estar en chip con otros componentes del descodificador de vídeo 30, o fuera de chip con respecto a esos componentes.

[196] Una memoria intermedia de imágenes codificadas (CPB) puede recibir y almacenar datos de vídeo codificados (por ejemplo, unidades NAL) de un flujo de bits. La unidad de descodificación por entropía 150 puede recibir datos de vídeo codificados (por ejemplo, unidades NAL) de la CPB y analizar las unidades NAL para descodificar elementos sintácticos. La unidad de descodificación por entropía 150 puede descodificar por entropía elementos sintácticos codificados por entropía en las unidades NAL.

[197] De acuerdo con aspectos de esta divulgación, la unidad de descodificación por entropía 150 puede configurarse para descodificar datos de paleta usando un código TEGk, como se describió anteriormente con respecto al ejemplo de la FIG. 1. En particular, de acuerdo con aspectos de esta divulgación, la unidad de descodificación por entropía 150 puede descodificar datos que indican una longitud de serie para una serie de índices de paleta (por ejemplo, una serie de índices que tienen el mismo valor o una serie de índices que se copian desde índices contiguos superiores) basándose en una longitud de serie máxima determinada. En algunos ejemplos, la unidad de descodificación por entropía 150 puede descodificar la longitud de serie usando un código TEG2.

[198] La unidad de procesamiento de predicción 152, la unidad de cuantificación inversa 154, la unidad de procesamiento de transformada inversa 156, la unidad de reconstrucción 158 y la unidad de filtro 160 pueden generar datos de vídeo descodificados basándose en los elementos sintácticos extraídos del flujo de bits. Las unidades NAL del flujo de bits pueden incluir unidades NAL de fragmentos codificados. Como parte de la descodificación del flujo de bits, la unidad de descodificación por entropía 150 puede extraer y descodificar por entropía elementos sintácticos de las unidades 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 con una imagen que contiene el fragmento.

[199] Además de los elementos sintácticos de descodificación del flujo de bits, el descodificador 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 descodificador 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 descodificador de vídeo 30 puede reconstruir bloques residuales de la CU.

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

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

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

[203] La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (RefPicList0) y una segunda lista de imágenes de referencia (RefPicList1) basándose en los elementos sintácticos extraídos del flujo de bits. Además, si se codifica una PU usando inter-predicción, la unidad de descodificación 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 uno o más bloques de referencia para la PU, bloques predictivos de luma, Cb y Cr para la PU.

[204] La unidad de reconstrucción 158 puede utilizar los bloques de transformada de luma, Cb y Cr asociados con las TU de una CU y los bloques de luma, Cb y Cr predictivos de las PU de la CU, es decir, datos de intra-predicción o 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 transformada de luma, Cb y Cr a las muestras correspondientes de los bloques de luma, Cb y Cr predictivos para reconstruir los bloques de codificación de luma, Cb y Cr de la CU.

[205] La unidad de filtro 160 puede realizar una operación de desbloqueo para reducir los artefactos de bloqueo asociados con los bloques de codificación de luma, Cb y Cr de la CU. El descodificador de vídeo 30 puede almacenar los bloques de codificación de luma, Cb y Cr de la CU en la memoria intermedia de imágenes descodificadas 162. La memoria intermedia de imágenes descodificadas 162 puede proporcionar imágenes de referencia para una posterior compensación de movimiento, intra-predicción y presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Por ejemplo, el descodificador de vídeo 30 puede realizar, basándose en los bloques de luma, Cb y Cr en la memoria intermedia de imágenes descodificadas 162, operaciones de intra-predicción o de inter-predicción en PU de otras CU.

[206] De acuerdo con diversos ejemplos de esta divulgación, el descodificador de vídeo 30 puede configurarse para realizar una codificación basada en paleta. La unidad de descodificación basada en paleta 165, por ejemplo,

puede realizar una descodificación basada en paleta cuando se selecciona un modo de descodificación basado en paleta, por ejemplo, para una CU o PU. Por ejemplo, la unidad de descodificación basada en paleta 165 puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxel, 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 píxel en la paleta basándose en la información, y reconstruir los valores de píxel del bloque basándose en los valores de píxel seleccionados en la paleta. Aunque se describen diversas funciones como realizadas por la unidad de descodificación basada en paleta 165, algunas o todas estas funciones pueden ser realizadas por otras unidades de procesamiento, o una combinación de diferentes unidades de procesamiento.

[207] La unidad de descodificación basada en paleta 165 puede recibir información de modo de codificación de paleta, y realizar las operaciones anteriores cuando la información del modo de codificación de paleta indica que el modo de codificación de paleta se aplica al bloque. Cuando la información del modo de codificación de paleta indica que el modo de codificación de paleta no se aplica al bloque, o cuando otra información de modo indica el uso de un modo diferente, el descodificador 30 puede descodificar el bloque de datos de vídeo usando un modo de codificación no basado en paleta, por ejemplo, un modo 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 con un proceso de codificación HEVC.

[208] De acuerdo con aspectos de esta divulgación, la unidad de descodificación basada en paleta 165 puede configurarse para realizar cualquier combinación de las técnicas para la codificación de paletas descritas en el presente documento. Por ejemplo, de acuerdo con los aspectos de esta divulgación, la unidad de descodificación basada en paleta 165 puede determinar un valor de un elemento sintáctico que indica, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra respectiva del bloque está codificada con un primer modo de paleta, donde el primer modo de paleta incluye codificar la muestra respectiva del bloque sin usar un índice en unos valores de paleta de colores para el bloque. Por ejemplo, la unidad de descodificación basada en paleta 165 puede determinar un valor de un elemento sintáctico a nivel de bloque que indique si alguna muestra del bloque se debe descodificar como una muestra de salida (por ejemplo, una muestra que no se puede reconstruir usando una entrada de color de la paleta). En algunos ejemplos, la unidad de descodificación basada en paleta 165 puede determinar un indicador de salida de un bit para un bloque que indica si se debe descodificar alguna muestra del bloque como una muestra de salida.

[209] De forma adicional o alternativa, la unidad de descodificación basada en paleta 165 puede determinar al menos uno de los datos que indica un tamaño de paleta máximo de unos valores de paleta de colores para codificar un bloque de datos de vídeo o datos que indican un tamaño de predictor de paleta máximo de un predictor de paleta determinar los valores de paleta de colores. Por ejemplo, la unidad de descodificación basada en paleta 165 puede descodificar tales datos de un conjunto de parámetros, tal como un SPS.

[210] De forma adicional o alternativa, la unidad de descodificación basada en paleta 165 puede determinar, para un píxel asociado con un índice de paleta que relaciona un valor del píxel con un valor de color en una paleta de colores para codificar el píxel, una longitud de serie de una serie de índices de paleta que se codifican junto con el índice de paleta del píxel (por ejemplo, una serie de índices que tienen el mismo valor o una serie de índices que se copian de los índices contiguos superiores). Es decir, la unidad de descodificación basada en paleta 165 puede determinar que un valor de índice para una muestra particular se descodifica con una serie de otros índices de paleta posteriores. La unidad de descodificación basada en paleta 165 también puede determinar una longitud de serie máxima para una serie máxima de índices de paleta que pueden descodificarse con el índice de paleta del píxel. Como se indicó anteriormente con respecto a la unidad de descodificación por entropía 150, el descodificador de vídeo 30 puede descodificar datos que indican la longitud de serie basándose en la longitud de serie máxima determinada.

[211] La FIG. 4 es un diagrama conceptual que ilustra un ejemplo de determinación de una paleta para codificar datos de vídeo, coherente 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 con las primeras paletas 184 y una segunda CU 188 que está asociada con las segundas paletas 192. Como se describe en mayor detalle a continuación y de acuerdo con 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 un modo de codificación de infra-predicción y el bloque 200 que está codificado con un modo de codificación de inter-predicción.

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

[213] En general, una paleta se refiere a un número de valores de píxel que son dominantes y/o representativos para una CU actualmente codificada, 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 con aspectos de esta divulgación, un codificador de vídeo (tal como un codificador de vídeo 20 o un descodificador de vídeo 30) puede codificar las 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.

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

[215] En el ejemplo de la FIG. 4, las primeras paletas 184 incluyen tres entradas 202-206 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 202-206 relacionan los índices de paleta con los valores de píxel, incluyendo el valor de píxel A, el valor de píxel B, el valor de píxel C y el valor de píxel D, respectivamente. Como se describe en el presente documento, en lugar de codificar los valores de píxel reales de la primera CU 180, un codificador de vídeo (como el codificador de vídeo 20 o el descodificador de vídeo 30) puede usar codificación basada en paleta para codificar los píxeles del bloque usando los índices de paleta 1-3. 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 con un valor de píxel en una o más de las primeras paletas 184. El descodificador de vídeo 30 puede obtener los índices de paleta de un flujo de bits y reconstruir los valores de píxel usando los índices de paleta 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 usar por parte del descodificador de vídeo 30 en descodificación basada en paleta.

[216] De acuerdo con aspectos de esta divulgación, se puede señalar un tamaño de paleta máximo para las primeras paletas 184. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican un tamaño de paleta máximo, por ejemplo, en términos del número de entradas que pueden incluirse en las primeras paletas 184. En algunos ejemplos, uno o más elementos sintácticos que indican el tamaño de paleta máximo (por ejemplo, MAX_PLT_SIZE) pueden incluirse en un SPS que está activo para CU 180. En otros ejemplos, uno o más elementos sintácticos que indican el tamaño de paleta máximo se pueden incluir en otro conjunto de parámetros, como un VPS o PPS, o en datos de cabecera tales como datos de cabecera de fragmento o datos asociados con una LCU o CU.

[217] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden variar, usando uno o más elementos sintácticos que indican el tamaño de paleta máximo; el tamaño de paleta máximo puede basarse en el perfil, nivel o profundidad de bits particular de los datos de vídeo que están siendo codificados. En otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden variar, usando uno o más elementos sintácticos que indican el tamaño de paleta máximo; el tamaño de paleta máximo puede basarse en un tamaño del bloque codificado, tal como CU 180.

[218] En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar los datos que indican un tamaño de paleta máximo cuando se construyen las primeras paletas 184 para CU 180. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden continuar agregando entradas a las primeras paletas 184 hasta alcanzar el tamaño de paleta máximo indicado por los datos. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden entonces codificar CU 180 usando las primeras paletas construidas 184.

[219] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar las segundas paletas 192 basándose en las primeras paletas 184. Por ejemplo, el codificador de vídeo 20 y/o el descodificador de vídeo 30 pueden ubicar uno o más bloques a partir de los cuales se determinan las paletas de predicción, en este ejemplo, las primeras paletas 184. La combinación de entradas que se utilizan con fines de predicción puede denominarse paleta de predictores.

[220] En el ejemplo de la FIG. 4, las segundas paletas 192 incluyen tres entradas 208-212 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 208-212 relacionan los índices de paleta 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 de predicción, aunque la paleta de predictores puede incluir entradas de un número de bloques) se incluyen en las segundas paletas 192.

[221] En el ejemplo de la FIG. 4, uno o más elementos sintácticos se ilustran como un vector 216. El vector 216 tiene una serie de bins (o bits) asociados, con cada bin que indica si el predictor de paleta asociado con ese bin se usa para predecir una entrada de la paleta actual. Por ejemplo, el vector 216 indica que las primeras dos 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.

[222] En algunos ejemplos, como se indicó anteriormente, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar un predictor de paleta (que también puede denominarse tabla de predictores de paleta o lista de predictores de paleta) cuando se realiza la predicción de paleta. El predictor de paleta puede incluir entradas de paletas de uno o más bloques contiguos que se usan para predecir una o más entradas de una paleta para codificar un bloque actual. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden construir la lista de la misma manera. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos (tales como el vector 216) para indicar qué entradas del predictor de paleta se van a copiar a una paleta para codificar un bloque actual.

[223] Por lo tanto, en algunos ejemplos, las entradas de paleta previamente descodificadas se almacenan en una lista para usar como un predictor de paleta. Esta lista puede usarse para predecir entradas de paleta en la CU del modo de paleta actual. Se puede señalar un vector de predicción binario en el flujo de bits para indicar qué entradas en la lista se reutilizan en la paleta actual. En la Solicitud Provisional de Estados Unidos n.º 62/018,461, presentada el 27 de junio de 2014, la codificación de longitud de serie se utiliza para comprimir el predictor del paleta binario. En un ejemplo, el valor de longitud de serie se codifica usando código Golomb exponencial de orden 0^o.

[224] De acuerdo con aspectos de esta divulgación, en algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 (por ejemplo, la unidad de codificación por entropía 118 y la unidad de descodificación por entropía 150) pueden configurarse para codificar (por ejemplo, codificar y descodificar, respectivamente) un vector binario de predicción de paleta para una paleta de un bloque que usa un código Golomb exponencial (TEGk) truncado de orden *k*-ésimo, como se describió anteriormente con respecto al ejemplo de la FIG. 1.

[225] En algunos casos, el codificador de vídeo 2.0 y el descodificador de vídeo 30 pueden configurarse para codificar el vector de predicción de paleta binaria usando el código TEGk junto con las técnicas descritas en el documento de presentación de norma Seregin et al., "Non-SCCE3: Run-Length Coding for Palette Predictor" [No SCCE3: Codificación de longitud de serie para predictor de paleta], JCTVC-RO228, Sapporo, JP, 30 de junio - 9 de julio de 2014(en adelante, JCTVC-R0228). En JCTVC-R0228, la codificación de longitud de serie se utiliza para codificar los elementos cero en un vector binario con las siguientes condiciones y pasos:

- El valor de longitud de serie igual a 1 indica el final de la predicción
- El final de la predicción no se señala para el último 1 en el vector binario
- El número de elementos cero precedentes se codifican para cada 1 en el vector binario
- Si el número de elementos cero es mayor que 0, se señala el número más uno, debido al valor de salida de 1
- El valor de longitud de serie se codifica utilizando código Golomb exponencial de orden 0

En un ejemplo con fines ilustrativos, un vector de predicción de paleta binaria puede ser igual a {1100100010000}, indicando que cuatro entradas (indicadas por las cuatro) del predictor de paleta se copian a la paleta para codificar un bloque actual. En este ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar el vector como 0-0-3-4-1.

[226] De acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar el vector de predicción de paleta binaria usando un valor de serie máximo *X* para el vector, que puede ser igual al número de entradas de paleta en la lista de predictores de paleta menos la posición actual en orden de escaneado menos uno). De acuerdo con un ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 usan un código TEG0 para codificar el valor de serie.

[227] Las técnicas de esta divulgación también se refieren a la señalización de un tamaño de predictor de paleta máximo. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar datos que indican un tamaño de predictor de paleta máximo, por ejemplo, en términos del número de bits que pueden incluirse en el vector 216. En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos que indican el tamaño de predictor de paleta máximo en relación con uno o más valores diferentes. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador 30 pueden configurarse para codificar datos que indican el tamaño de predictor de paleta máximo como una delta (por ejemplo, diferencia) entre el tamaño de predictor de paleta máximo y el tamaño de paleta máximo. En algunos casos, el codificador de vídeo y el descodificador de vídeo 30 pueden codificar el delta usando al menos uno de un código de longitud fija, un código Golomb-Rice o un código Golomb exponencial.

[228] En algunos ejemplos, los elementos sintácticos que indican el tamaño de predictor de paleta máximo se pueden incluir en un SPS. En otros ejemplos, los elementos sintácticos que indican el tamaño de predictor de paleta máximo se pueden incluir en otro conjunto de parámetros, como un VPS o PPS, o en datos de cabecera tales como datos de cabecera de fragmento o datos asociados con una LCU o CU.

[229] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden variar, utilizando los elementos sintácticos que indican el tamaño de predictor de paleta máximo: el tamaño de predictor de paleta máximo puede basarse en el perfil, nivel o profundidad de bits particular de los datos de vídeo codificados. En otros ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden variar, utilizando los elementos sintácticos que

indican el tamaño de predictor de paleta máximo; el tamaño de predictor de paleta máximo puede basarse en un tamaño del bloque codificado.

[230] En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar los datos relativos al tamaño de predictor de paleta máximo al construir las segundas paletas 192 para codificar CU 188. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden continuar agregando entradas a una paleta de predictores (por ejemplo, y bits al vector 216) hasta alcanzar un tamaño de predictor de paleta máximo, como lo indican los datos. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden entonces usar el vector 216 para construir las segundas paletas 192 para la CU 188.

[231] La FIG. 5 es un diagrama conceptual que ilustra un ejemplo de determinación de índices de paleta para una paleta para un bloque de píxeles, coherente con las técnicas de esta divulgación. Por ejemplo, la FIG. 5 incluye un mapa 240 de índices de paleta que relacionan posiciones respectivas de píxeles asociados con los índices de paleta con una entrada de paletas 244. Por ejemplo, el índice 1 está asociado con el Valor A, el índice 2 está asociado con el Valor B, y el índice 3 está asociado con el Valor C. Además, cuando se indican muestras de salida usando señalización de salida implícita, el codificador de vídeo 20 y el descodificador de vídeo 30 también pueden añadir un índice adicional a las paletas 244, ilustradas en la FIG. 5 como índice 4, que puede indicar que las muestras del mapa 240 asociadas con el índice 4 son muestras de salida. En este caso, el codificador de vídeo 20 puede codificar (y el descodificador de vídeo 30 puede obtener, a partir de un flujo de bits codificado) una indicación de un valor de píxel real (o su versión cuantificada) para una posición en el mapa 240 si el valor de píxel no está incluido en las paletas 244.

[232] En algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar un mapa adicional que indique qué posiciones de píxel están asociadas con los índices de paleta. Por ejemplo, suponga 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.

[233] El codificador de vídeo 20 puede, en dicho ejemplo, codificar también un índice de paleta (mostrado en el ejemplo de la FIG. 5 como valores 1-3) para indicar ese valor de píxel en la paleta y para permitir que el descodificador de vídeo reconstruya el valor de píxel.

[234] 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 usar por el descodificador de vídeo 30 al reconstruir el valor de píxel. En algunos casos, el valor de píxel puede codificarse de una manera con pérdidas.

[235] El valor de un píxel en una posición de una CU puede proporcionar una indicación de valores de uno o más píxeles diferentes 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 asignar al mismo valor de índice (en el caso de codificación con pérdida, en la que se puede asignar más de un valor de píxel a un único valor de índice).

[236] En consecuencia, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican un número de píxeles consecutivos o índices de paleta en un orden de escaneado dado que están codificados juntos. Como se indicó anteriormente, la secuencia de índices de paleta (o valores de píxel indicados por los índices de paleta) se puede denominar aquí una serie. El descodificador de vídeo 30 puede obtener los elementos sintácticos que indican una serie desde un flujo de bits codificado y usar los datos para determinar el número de ubicaciones consecutivas que tienen el mismo valor de píxel o índice.

[237] Como se indicó anteriormente, las series se pueden usar junto con un modo CopyFromTop o de valor. En un ejemplo con fines ilustrativos, considere las filas 264 y 268 del mapa 240. Suponiendo una dirección de escaneado horizontal de izquierda a derecha, la fila 264 incluye tres índices de paleta de "1", dos índices de paleta de "2" y tres índices de paleta de "3". La fila 268 incluye cinco índices de paleta de "1", dos índices de paleta 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 salida de nivel de muestra para la señalización de salida explícita), que se puede denominar como una muestra de salida.

[238] En este ejemplo, el codificador de vídeo 20 puede usar el modo CopyFromTop 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 serie de dos entradas consecutivas en la dirección de exploración en la fila 268 es la misma que la primera posición de la fila 264.

[239] Después de codificar el uno o más elementos sintácticos que indican la primera posición de la fila 264 y la serie de dos entradas (anotadas anteriormente), el codificador de vídeo 20 puede codificar las posiciones cuarta y quinta en la fila 268 (de izquierda a derecha), usando el modo 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 serie de 1 (por ejemplo, modo de valor). Por lo tanto, el codificador de vídeo 20 codifica estas dos posiciones sin referencia a otra línea.

[240] El codificador de vídeo 20 puede entonces codificar la primera posición que tiene un valor de índice de 3 en la fila 268 usando el modo CopyFromTop con respecto a la fila superior 264. Por ejemplo, el codificador de vídeo 20 puede señalar un modo CopyFromTop y una serie de 1. Por consiguiente, el codificador de vídeo 20 puede seleccionar entre valores de píxel de codificación o índices de paleta de una línea con relación a otros valores de la línea, por ejemplo, usando una serie, codificando valores de píxel o de una línea con relación a valores de otra línea (o columna), o una combinación de las mismas. El codificador de vídeo 20 puede, en algunos ejemplos, realizar una optimización de velocidad/distorsión para realizar la selección.

[241] El codificador de vídeo 20 puede entonces codificar la muestra de salida 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 salida. Es decir, el codificador de vídeo 20 puede codificar una indicación de que la posición final de la fila 268 es una muestra de salida (por ejemplo, índice 4), así como una indicación del valor de muestra. El descodificador de vídeo 30 puede obtener la sintaxis descrita anteriormente a partir de un flujo de bits codificado y reconstruir la fila 268 usando tal sintaxis.

[242] Como se señaló anteriormente, puede haber dos técnicas para codificar la muestra de salida. Por ejemplo, con señalización de salida explícita, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar un indicador de modo de salida explícito por muestra 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 salida, el codificador de vídeo 20 y el descodificador 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 salida, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos adicionales para indicar si el modo es CopyFromTop o de valor, tal como un indicador de SPoint.

[243] Con la señalización de salida implícita, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden agregar un índice adicional a las paletas 244 (índice de entrada 4). El codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar el índice adicional para las paletas 244 para indicar que una muestra se codifica como una muestra de salida, por ejemplo, el índice 4. El índice adicional, sin embargo, no tiene un valor de color asociado. Más bien, el codificador de vídeo 20 y el descodificador de vídeo 30 también codifican valores de color para cada muestra que está asociada con el índice adicional. Si la muestra no está codificada como una muestra de salida, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar datos para indicar si el modo es CopyFromTop o de valor, como un indicador de SPoint.

[244] De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para codificar uno o más elementos sintácticos a nivel de bloque que indican, para todas las muestras de un bloque de datos de vídeo, si al menos una muestra del bloque está codificada basándose en un valor de color que no se incluye en una paleta de colores para el bloque. Con respecto al ejemplo de la FIG. 5, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar uno o más elementos sintácticos asociados con el mapa 240 que indican que al menos una muestra del mapa 240 está codificada como una muestra de salida, es decir, la muestra final de la fila 268.

[245] En un ejemplo, el uno o más elementos sintácticos pueden ser un indicador de salida a nivel de bloque (al que se hace referencia a continuación como simplemente "indicador de salida"). Por ejemplo, el codificador de vídeo 20 puede codificar un indicador de salida que tiene un valor de uno para indicar que el mapa 240 incluye una muestra codificada como una muestra de salida. Asimismo, el descodificador de vídeo 30 puede descodificar un indicador de salida que tiene un valor de uno, que indica que el mapa 240 incluye una muestra codificada como una muestra de salida. En consecuencia, el codificador de vídeo 20 puede codificar y el descodificador de vídeo 30 puede descodificar el mapa 240 de acuerdo con el indicador de salida. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden agregar el índice 4 a las primeras paletas 244, que pueden usarse para representar muestras codificadas como muestras de salida. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar este índice adicional durante la codificación del mapa 240.

[246] De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para omitir la codificación de cierta sintaxis basándose en el indicador de salida y el tamaño de la paleta que se usa para codificar un bloque en particular. Es decir, mientras el ejemplo de la FIG. 5 ilustra las primeras paletas 244 que tienen tres entradas, en algunos casos, una paleta para codificar un bloque de datos de vídeo puede

incluir una sola entrada. En tales casos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para omitir la codificación de cierta sintaxis relacionada con la salida basándose en la paleta que tiene una sola entrada y el indicador de salida que indica que no hay muestras del bloque codificadas como muestras de salida.

[247] Por ejemplo, como en casos en que el indicador de salida indica que ninguna muestra del bloque son muestras de salida y cuando el tamaño de una paleta es uno, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para inferir que todas las muestras del bloque tienen el mismo valor de índice (por ejemplo, la única entrada del codificador de vídeo 20 y el descodificador de vídeo 30 de la paleta también puede, por lo tanto, omitir la codificación de todos los demás datos usados para determinar los índices de paleta del bloque.

[248] En el ejemplo anterior, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar explícitamente la sintaxis de salida a nivel de bloque (por ejemplo, el codificador de vídeo 20 puede codificar un indicador de salida en el flujo de bits, y el descodificador 30 puede descodificar dicho indicador del flujo de bits). Sin embargo, en algunos ejemplos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden inferir (por ejemplo, determinar, sin codificar ni descodificar el elemento sintáctico mencionado anteriormente) el valor del elemento sintáctico de salida a nivel de bloque basándose en el tamaño de paletas 244 utilizado para mapa de código 240. Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden hacer una determinación preliminar con respecto al tamaño de la paleta para determinar el valor del indicador de salida a nivel de bloque. El elemento sintáctico de salida a nivel de bloque solo se puede codificar en el flujo de bits cuando el tamaño de la paleta es mayor que cero.

[249] De acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar inicialmente un tamaño de una paleta para codificar un bloque de datos de vídeo. Basándose en que el tamaño de la paleta sea cero, el codificador de vídeo 20 y el descodificador 30 pueden determinar que el indicador de salida es igual a uno y que todas las muestras del bloque están codificadas como muestras de salida, porque no hay otras entradas disponibles para codificar muestras.

[250] Por ejemplo, como en casos en que el tamaño de la paleta es cero, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden configurarse para determinar automáticamente que todas las muestras del bloque tienen el mismo valor de índice (por ejemplo, la entrada adicional de la paleta asociada con muestras de salida). El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden, por lo tanto, omitir la codificación del indicador de salida así como también todos los otros datos usados para determinar los índices de paleta del bloque.

[251] La FIG. 6 es un diagrama conceptual que ilustra un ejemplo de determinación de la longitud de serie máxima para el modo CopyFromAbove, suponiendo un orden de escaneado de trama, coherente con las técnicas de esta divulgación. Como se indicó anteriormente, las técnicas de esta divulgación incluyen la sintaxis de codificación para la codificación de paleta utilizando un código que representa un valor potencial máximo de la sintaxis que se codifica.

[252] En un ejemplo con fines ilustrativos, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden codificar una longitud de serie de una serie de índices de paleta (por ejemplo, una serie de índices que tienen el mismo valor o una serie de índices que se copian desde índices contiguos superiores). Por ejemplo, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar, para una muestra codificada en una paleta actual, una longitud de serie de una serie de índices de paleta que se codifica junto con la muestra actual. El codificador de vídeo 20 y el descodificador de vídeo 30 también pueden determinar una longitud de serie máxima para una serie máxima de índices de paleta que pueden codificarse con el índice de paleta del píxel. El codificador de vídeo 20 y el descodificador de vídeo 30 pueden entonces codificar datos que indican la longitud de serie basándose en la longitud de serie máxima determinada.

[253] En algunos casos, de acuerdo con aspectos de esta divulgación, la sintaxis puede codificarse usando una forma de código Golomb exponencial, tal como el código de TEGk descrito en el presente documento. Para usar el código TEGk, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden determinar una longitud de serie máxima como el número de píxeles en la CU actual menos la posición actual en orden de escaneado menos 1.

[254] En algunas técnicas de codificación de paleta, series de píxeles asociadas con el modo CopyFromAbove (donde un codificador de vídeo copia un índice de un píxel por encima del píxel actual), no se permite incluir ningún píxel de salida. Es decir, un codificador de vídeo debe detener una serie de CopyFromAbove si el píxel contiguo superior del píxel actual es un píxel codificado como muestra de salida. Por lo tanto, la longitud de serie máxima de CopyFromAbove está limitada por la distancia entre la posición de píxel actual y la posición que tiene un píxel contiguo superior que se escapa en el orden de escaneado.

[255] En un ejemplo con fines ilustrativos, la posición de inicio de una serie de CopyFromAbove en orden de escaneado es A, el píxel contiguo superior al píxel en la posición $A+L$ ($L > 0$) (o en algunos ejemplos, $A+L$ ($Z > 1$)) se codifica como una muestra de salida, y el píxel contiguo superior al píxel en la posición $A+1$ ($L < 1$) no está codificado como una muestra de salida. Si tal píxel L no existe, el codificador de vídeo 20 puede asignar L a la posición después del último píxel en el bloque en orden de exploración. De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden usar TEGk para codificar la longitud de serie para el modo CopyFromAbove con la restricción de que la longitud de serie máxima codificada no sea mayor que $L-1$. De forma

alternativa, si se utilizan prefijos unarios mayores que 0, mayores que 1 y mayores que 2 cuando se codifica una longitud de serie, el codificador de vídeo 20 o el descodificador de vídeo 30 pueden establecer la longitud de serie máxima de la serie del mapa de índice para ser codificado usando TEGk a L-4.

[256] En casos en los que el descodificador de vídeo 30 o el codificador de vídeo 20 no pueden determinar si un píxel en una posición corresponde a un píxel de salida o no, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden procesar el píxel como si no estuviera codificado como una muestra de salida, es decir la longitud de serie máxima codificada no se detiene. En el ejemplo de la FIG. 6, si ninguno de los píxeles abarcados por las líneas discontinuas 280 está codificado como una muestra de salida, la longitud de serie máxima posible es 35 (es decir, el número de posiciones de píxeles sin sombrear). Si uno o más de los píxeles dentro de las líneas discontinuas 280 están codificados como una muestra de salida, suponiendo que el píxel marcado como el píxel de salida (la posición de píxel con la "X") es el primer píxel de salida dentro de las líneas discontinuas 280 en orden de escaneado, entonces la copia codificada máxima posible sobre la longitud de serie es cinco.

[257] En algunos ejemplos, el descodificador de vídeo 30 solo puede determinar el modo de serie (por ejemplo, el modo de paleta en el que se codifican los píxeles) para los píxeles dentro de las líneas discontinuas 280. Por lo tanto, en el peor de los casos, el descodificador de vídeo 30 realiza la determinación para píxeles de Ancho de Bloque-1. En algunos ejemplos, el descodificador de vídeo 30 puede configurarse para implementar ciertas restricciones con respecto al número máximo de píxeles para los que se comprueba el modo de serie. Por ejemplo, el descodificador de vídeo 30 solo puede comprobar los píxeles dentro de las líneas discontinuas 280 si los píxeles están en la misma fila que el píxel actual. El descodificador de vídeo 30 puede inferir que todos los demás píxeles dentro de las líneas discontinuas 280 no están codificados como muestras de salida. El ejemplo de la FIG. 6 supone un orden de escaneado de trama. Sin embargo, las técnicas se pueden aplicar a otros órdenes de escaneado, como vertical, horizontal transversal y vertical transversal.

[258] La FIG. 7 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basándose en uno o más elementos sintácticos a nivel de bloque que indican si cualquier muestra de un bloque está codificada como muestra de salida, de acuerdo con las técnicas de esta divulgación. El proceso de la FIG. 7 se describe en general como realizado por el codificador de vídeo 20 con fines ilustrativos, aunque una variedad de otros procesadores también pueden llevar a cabo el proceso mostrado en la FIG. 7.

[259] En el ejemplo de la FIG. 7, el codificador de vídeo 20 determina una paleta para codificar un bloque de datos de vídeo (300). En algunos ejemplos, el codificador de vídeo 20 puede determinar las paletas basadas en paleta de uno o más bloques previamente codificados, por ejemplo, usando un predictor de paleta. El codificador de vídeo 20 también puede determinar el tamaño de la paleta (302). Por ejemplo, el codificador de vídeo 20 puede determinar un número de entradas en la paleta determinada.

[260] El codificador de vídeo 20 puede determinar si el tamaño de la paleta es cero (304). Basándose en que el tamaño de paleta sea igual a cero (la rama sí del paso 304), el codificador de vídeo 20 puede determinar la sintaxis de salida a nivel de bloque que indica que al menos una muestra del bloque es una muestra de salida (306). En un ejemplo, el codificador de vídeo 20 puede determinar que un indicador de salida a nivel de bloque es igual a uno. El codificador de vídeo 20 codifica entonces todas las muestras del bloque como muestras de salida y sin codificar otros elementos sintácticos para el bloque (308). Por ejemplo, el codificador de vídeo 20 puede no codificar una indicación del indicador de salida a nivel de bloque, porque el elemento sintáctico de salida a nivel de bloque solo puede codificarse en el flujo de bits cuando el tamaño de la paleta es mayor que cero. Además, el codificador de vídeo 20 puede no codificar otros datos para los índices de paleta del bloque, como la sintaxis que indica un modo de paleta (por ejemplo, de valor o CopyFromTop), la sintaxis asociada con la determinación de series, la sintaxis asociada con los índices de paleta y cualquier otra sintaxis relacionada.

[261] Si el tamaño de la paleta no es cero (la rama no del paso 304), el codificador de vídeo 20 puede determinar si alguna muestra del bloque está codificada como muestra de salida (310). Basándose en determinar que al menos una muestra del bloque está codificada como una muestra de salida (la rama sí del paso 310) el codificador de vídeo 20 puede determinar la sintaxis de salida a nivel de bloque que indica que al menos una muestra del bloque está codificada como una muestra de salida y codificar una indicación de la sintaxis a nivel de bloque en un flujo de bits con el bloque (312). Por ejemplo, el codificador de vídeo 20 puede establecer un indicador de salida a nivel de bloque igual a uno y codificar una indicación del indicador de salida en el flujo de bits. El codificador de vídeo 20 también codifica el bloque con modos de codificación de paleta, que incluye la codificación de al menos un píxel del bloque como una muestra de salida (314).

[262] Basándose en que no se codifiquen muestras como muestras de salida (la rama no del paso 310), el codificador de vídeo 20 puede determinar si el tamaño de la paleta de la paleta para el bloque es igual a uno (316). Basándose en determinar que el tamaño de la paleta no es igual a uno (la rama no del paso 316), el codificador de vídeo 20 puede determinar la sintaxis de salida a nivel de bloque que indica que ninguna muestra del bloque está codificada como muestras de salida, codificar una indicación de sintaxis de salida a nivel de bloque en el flujo de bits (318). Por ejemplo, el codificador de vídeo 20 puede establecer un indicador de salida a nivel de bloque igual a cero y codificar una indicación del indicador de salida en el flujo de bits. El codificador de vídeo 20 también codifica el bloque

usando modos de codificación de paleta pero sin codificar ninguna muestra de salida (320). Por ejemplo, el codificador de vídeo 20 puede codificar los índices de paleta del bloque usando los modos CopyFromTop o de valor y codificar la sintaxis asociada con el uso de dichos modos, por ejemplo, la sintaxis que indica modos, índices de paleta, series y similares.

[263] Basándose en determinar que el tamaño de la paleta es igual a uno (la rama sí del paso 316), el codificador de vídeo 20 puede determinar la sintaxis de salida a nivel de bloque que indica que ninguna muestra del bloque está codificada como muestra de salida y codificar una indicación de la sintaxis de salida a nivel de bloque en el flujo de bits (322). Por ejemplo, el codificador de vídeo 20 puede establecer un indicador de salida a nivel de bloque igual a cero y codificar una indicación del indicador de salida en el flujo de bits. El codificador de vídeo 20 también codifica una indicación de que todas las muestras del bloque tienen el mismo valor de índice y sin codificación de otra sintaxis (324). Por ejemplo, el codificador de vídeo 20 puede omitir la codificación de la sintaxis asociada con el uso de los modos de paleta, por ejemplo, la sintaxis que indica modos, índices de paleta, series y similares.

[264] La FIG. 8 es un diagrama de flujo que ilustra un proceso de ejemplo para descodificar un bloque de datos de vídeo basado en uno o más elementos sintácticos a nivel de bloque que indican si se descodifican muestras de un bloque como muestras de salida, de acuerdo con las técnicas de esta divulgación. El proceso de la FIG. 8 se describe en general como realizado por el descodificador de vídeo 30 con fines ilustrativos, aunque una variedad de otros procesadores también pueden llevar a cabo el proceso mostrado en la FIG. 8.

[265] En el ejemplo de la FIG. 8, el descodificador de vídeo 30 determina una paleta para descodificar un bloque de datos de vídeo (340). En algunos ejemplos, el descodificador de vídeo 30 puede determinar las paletas basadas en paleta de uno o más bloques previamente codificados, por ejemplo, usando un predictor de paleta señalado en un flujo de bits que se está descodificando. El descodificador de vídeo 30 también puede determinar el tamaño de la paleta (342). Por ejemplo, el descodificador de vídeo 30 puede determinar un número de entradas en la paleta determinada.

[266] El descodificador de vídeo 30 puede determinar si el tamaño de la paleta es cero (344). Basándose en que el tamaño de paleta sea igual a cero (la rama sí del paso 344), el descodificador de vídeo 30 puede determinar la sintaxis de salida a nivel de bloque que indica que al menos una muestra del bloque está codificada como muestra de salida (346). En un ejemplo, el descodificador de vídeo 30 puede determinar que un indicador de salida a nivel de bloque es igual a uno. Por ejemplo, el descodificador de vídeo 30 puede inferir que el indicador de salida a nivel de bloque es igual a uno sin descodificar el indicador del flujo de bits, porque el elemento sintáctico de salida a nivel de bloque solo puede codificarse en el flujo de bits cuando el tamaño de paleta es mayor que cero. El descodificador de vídeo 30 descodifica todas las muestras del bloque usando el color asociado con la muestra de salida y sin descodificar otros elementos sintácticos para el bloque (por ejemplo, que no sean los valores de color asociados con la muestra de salida) (348). Por ejemplo, como se indicó anteriormente, el descodificador de vídeo 30 puede no descodificar una indicación del indicador de salida a nivel de bloque. Además, el descodificador de vídeo 30 puede no descodificar otros datos para índices de paleta del bloque, como sintaxis que indica un modo de paleta (por ejemplo, valor o CopyFromTop), sintaxis que indica valores de índice, sintaxis asociada con la determinación de series y cualquier otra sintaxis relacionada.

[267] Si el tamaño de la paleta no es cero (la rama no del paso 344), el descodificador de vídeo 30 puede descodificar la sintaxis de salida a nivel de bloque del flujo de bits y determinar un valor de la sintaxis a nivel de bloque (350). Por ejemplo, el descodificador de vídeo 30 puede descodificar un indicador de salida a nivel de bloque y determinar si el valor es igual a cero o uno.

[268] El descodificador de vídeo 30 puede entonces determinar si alguna muestra del bloque está codificada como una muestra de salida, por ejemplo, basándose en la sintaxis descodificada (352). Basándose en determinar que al menos una muestra del bloque está codificada como una muestra de salida (la rama sí del paso 352), el descodificador de vídeo 30 puede descodificar el bloque con modos de codificación de paleta, incluida la descodificación de al menos una muestra del bloque como muestra de salida (354). El descodificador de vídeo 30 también puede descodificar al menos un valor de color correspondiente a las muestras de salida.

[269] Basándose en que no hay muestras codificadas como muestras de salida (la rama no del paso 352), el descodificador de vídeo 30 puede determinar si el tamaño de la paleta de la paleta para el bloque es igual a uno (356). Basándose en determinar que el tamaño de la paleta no es igual a uno (la rama no del paso 356), el descodificador de vídeo 30 puede descodificar el bloque usando modos de codificación de paleta pero sin descodificar ninguna muestra de salida (358). Por ejemplo, el descodificador de vídeo 30 puede descodificar índices de paleta del bloque usando los modos CopyFromTop o de valor y la sintaxis de descodificación asociada con el uso de tales modos, por ejemplo, sintaxis que indica modos, índices de paleta, series, y similares.

[270] Basándose en determinar que el tamaño de paleta es igual a uno (la rama sí del paso 356), el descodificador 30 puede descodificar el bloque usando la entrada de paleta de la paleta (por ejemplo, la única entrada presente en la paleta) y sin descodificar otra sintaxis (360). Por ejemplo, el descodificador de vídeo 30 puede omitir la

descodificación de la sintaxis asociada con el uso de los modos de paleta, por ejemplo, la sintaxis que indica modos, índices de paleta, series, y similares.

[271] La FIG. 9 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basado en uno o más elementos sintácticos que indican un tamaño de paleta máximo y un tamaño de predictor de paleta máximo, coherente con las técnicas de esta divulgación. El proceso de la FIG. 9 se describe en general como realizado por el codificador de vídeo 20 con fines ilustrativos, aunque una variedad de otros procesadores también pueden llevar a cabo el proceso mostrado en la FIG. 9.

[272] En el ejemplo de la FIG. 9, el codificador de vídeo 20 puede determinar un tamaño máximo de una paleta para codificar un bloque actual de datos de vídeo en el modo de paleta (380). Por ejemplo, el codificador de vídeo 20 puede configurarse para determinar un tamaño de paleta máximo basándose en una característica de los datos de vídeo que se codifican. En algunos ejemplos, el codificador de vídeo 20 puede determinar un tamaño de paleta máximo basado en una profundidad de bits de los datos (por ejemplo, una profundidad de bits de entrada o profundidad de bits de perfil), un tamaño de bloque del bloque, un perfil o nivel asociado con los datos de vídeo, o similares.

[273] El codificador de vídeo 20 también puede determinar un tamaño de predictor de paleta máximo de un predictor de paleta para crear una paleta del bloque actual (382). Por ejemplo, el codificador de vídeo 20 puede configurarse para determinar un tamaño de predictor de paleta máximo basado en una característica de los datos de vídeo que se codifican. En algunos ejemplos, el codificador de vídeo 20 puede determinar un tamaño de predictor de paleta máximo basado en una profundidad de bits de los datos (por ejemplo, una profundidad de bits de entrada o profundidad de bits de perfil), un tamaño de bloque del bloque, un perfil o nivel asociado con los datos de vídeo, o similares.

[274] El codificador de vídeo 20 también codifica datos que indican el tamaño de paleta máximo y/o el tamaño de predictor de la paleta máximo en un flujo de bits que incluye el bloque actual (386). En algunos ejemplos, el codificador de vídeo 20 puede codificar datos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo en relación con uno o más valores diferentes. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 puede configurarse para codificar datos que indican el tamaño de predictor de paleta máximo como una delta (por ejemplo, diferencia) entre el tamaño de predictor de paleta máximo y el tamaño de paleta máximo.

[275] De acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo en un SPS. En otros ejemplos, el codificador de vídeo 20 puede codificar tal sintaxis en otro conjunto de parámetros (por ejemplo, un PPS), en una cabecera de fragmento de un fragmento que incluye el bloque actual, o en cualquier otro lugar en el flujo de bits.

[276] El codificador de vídeo 20 también codifica el bloque actual de acuerdo con los datos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo (388). Por ejemplo, el codificador de vídeo 20 puede determinar una paleta que está limitada por el tamaño de paleta máximo y/o un predictor de paleta que está limitado por el tamaño de predictor de paleta máximo.

[277] La FIG. 10 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar un bloque de datos de vídeo basado en uno o más elementos sintácticos que indican un tamaño de paleta máximo y un tamaño de predictor de paleta máximo, coherente con las técnicas de esta divulgación. El proceso de la FIG. 10 se describe en general como realizado por el descodificador de vídeo 30 con fines ilustrativos, aunque una variedad de otros procesadores también pueden llevar a cabo el proceso mostrado en la FIG. 10.

[278] En el ejemplo de la FIG. 10, el descodificador de vídeo 30 descodifica datos que indican un tamaño de paleta máximo y/o un tamaño de predictor de paleta máximo a partir de un flujo de bits que incluye un bloque actual que se descodifica en el modo de paleta (400). En algunos ejemplos, el codificador de vídeo 20 puede codificar datos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo en relación con uno o más valores diferentes. Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo 20 puede configurarse para codificar datos que indican el tamaño de predictor de paleta máximo como una delta (por ejemplo, diferencia) entre el tamaño de predictor de paleta máximo y el tamaño de paleta máximo.

[279] De acuerdo con aspectos de esta divulgación, el descodificador de vídeo 30 puede descodificar uno o más elementos sintácticos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo desde un SPS. En otros ejemplos, el descodificador de vídeo 30 puede descodificar tal sintaxis de otro conjunto de parámetros (por ejemplo, un PPS), desde una cabecera de fragmento de un fragmento que incluye el bloque actual, o en cualquier otro lugar en el flujo de bits.

[280] El descodificador de vídeo 30 puede determinar el tamaño máximo de una paleta para descodificar el bloque actual basándose en los datos descodificados (402). El descodificador de vídeo 30 también puede determinar un tamaño de predictor de paleta máximo de un predictor de paleta para crear una paleta para el bloque actual basándose en los datos (404). El descodificador de vídeo 30 también descodifica el bloque actual de acuerdo con los datos que indican el tamaño de paleta máximo y/o el tamaño de predictor de paleta máximo (408). Por ejemplo, el descodificador

de vídeo 30 puede determinar una paleta que está limitada por el tamaño de paleta máximo y/o un predictor de paleta que está limitado por el tamaño de predictor de paleta máximo.

[281] La FIG. 11 es un diagrama de flujo que ilustra un proceso de ejemplo para codificar (codificar o decodificar) datos que indican una longitud de serie de una serie de píxeles basada en una longitud de serie máxima potencial, coherente con las técnicas de esta divulgación. El proceso de la FIG. 11 se describe en general como realizado por un codificador de vídeo, tal como el codificador de vídeo 20 o el decodificador de vídeo 30, con fines ilustrativos, aunque una variedad de otros procesadores también pueden llevar a cabo el proceso mostrado en la FIG. 11.

[282] En el ejemplo de la FIG. 11, el codificador de vídeo puede determinar un modo de paleta para codificar un píxel actual (420). Por ejemplo, el codificador de vídeo puede determinar si el píxel actual está codificado utilizando un modo CopyFromTop, un modo de valor u otro modo de codificación basado en paleta. El codificador de vídeo también determina una longitud de serie de una serie para el píxel actual (422). Por ejemplo, el codificador de vídeo determina la cantidad de índices de paleta que se codifican con el índice de paleta del píxel actual.

[283] El codificador de vídeo también determina una longitud de serie máxima para la serie (424). Por ejemplo, el codificador de vídeo puede determinar una longitud de serie máxima para una serie máxima de índices de paleta que pueden codificarse con el índice de paleta del píxel actual. En un ejemplo, el codificador de vídeo puede determinar una cantidad de píxeles en el bloque de datos de vídeo que incluye el píxel. El codificador de vídeo también puede determinar una posición del píxel actual en el bloque basándose en un orden de exploración utilizado para escanear los índices de paleta. El codificador de vídeo determina a continuación la longitud de serie máxima como el número de píxeles en el bloque menos la posición de píxel del píxel actual menos uno.

[284] El codificador de vídeo también codifica los datos que indican la longitud de serie de la serie basándose en la longitud de serie máxima determinada (426). Por ejemplo, de acuerdo con aspectos de esta divulgación, el codificador de vídeo puede codificar datos que indican la longitud de serie usando un código TEGk.

[285] Debe 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 con un codificador de vídeo.

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

[287] 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 denominarse en general codificador de vídeo. Del mismo modo, la codificación de vídeo puede referirse a la codificación de vídeo o a la decodificación de vídeo, según sea aplicable.

[288] 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 deben limitarse a estas combinaciones de ejemplos y pueden abarcar cualquier combinación concebible de los diversos aspectos de las técnicas descritas en esta divulgación.

[289] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de estos. 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 unos medios de almacenamiento de datos o unos medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) unos 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.

[290] A modo de ejemplo, y no de manera limitativa, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de

disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe apropiadamente 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 emiten datos magnéticamente, mientras que otros discos emiten datos ópticamente con láseres. Las combinaciones de los anteriores deberían incluirse también dentro del alcance de los medios legibles por ordenador.

[291] 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), matrices lógicas programables sobre el terreno (FPGA) u otros circuitos lógicos integrados o discretos equivalentes. 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 módulos de hardware y/o software dedicados 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.

[292] Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, que incluyen un teléfono 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.

[293] 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 codificar al menos uno de los datos, dentro de un conjunto de parámetros de secuencia (SPS) de un flujo de bits que contiene el bloque de datos de vídeo, que indica un tamaño máximo de paleta de una paleta de valores de color (184) para codificar un bloque de datos de vídeo (180) o datos que indican el tamaño máximo del predictor de paleta de un predictor de paleta para determinar la paleta de valores de color; y
 - 10 codificar el bloque de datos de vídeo de acuerdo con los datos.
2. El procedimiento de la reivindicación 1, en el que codificar los datos que indican el tamaño máximo de la paleta comprende codificar un elemento de sintaxis `palette_max_size` dentro del SPS.
- 15 3. El procedimiento de la reivindicación 1, en el que codificar los datos que indican el tamaño máximo de predictor de paleta del predictor de paleta comprende codificar los datos dentro del SPS del flujo de bits.
4. El procedimiento de la reivindicación 1, en el que codificar los datos que indican el tamaño máximo del predictor de paleta comprende datos de codificación que indican una diferencia entre el tamaño máximo del predictor de paleta y el tamaño máximo de la paleta.
- 20 5. El procedimiento de la reivindicación 4, en el que codificar los datos que indican la diferencia entre el tamaño máximo del predictor de paleta y el tamaño máximo de la paleta comprende codificar los datos que indican la diferencia entre el tamaño máximo del predictor de paleta y el tamaño máximo de la paleta usando al menos uno de un código de longitud fija, un código de Golomb-Rice o un código exponencial de Golomb.
- 25 6. El procedimiento de la reivindicación 1, en el que el bloque de datos de vídeo comprende un primer bloque de datos de vídeo, comprendiendo además el procedimiento:
 - 30 codificar segundos datos que indican un segundo tamaño máximo de paleta de una segunda paleta de valores de color (192) para codificar un segundo bloque de datos de vídeo (188), en el que el segundo tamaño máximo de paleta se basa en un tamaño del segundo bloque de datos de vídeo.
- 35 7. El procedimiento de la reivindicación 1, en el que codificar los datos que indican el tamaño máximo de la paleta comprende codificar uno o más elementos de sintaxis que indican el tamaño máximo de la paleta utilizando al menos uno de un código de longitud fija, un código de Golomb Rice o un código exponencial de Golomb.
8. El procedimiento de la reivindicación 1, en el que codificar los datos que indican el tamaño máximo de predictor de paleta comprende codificar uno o más elementos de sintaxis que indican el tamaño máximo del predictor de paleta utilizando al menos uno de un código de longitud fija, un código de Golomb Rice o un código exponencial de Golomb.
- 40 9. El procedimiento de la reivindicación 1, que comprende además:
 - 45 codificar uno o más elementos de sintaxis que indican que al menos un píxel del bloque se codifica utilizando un modo de paleta de escape; y
 - aumentar el tamaño máximo de la paleta indicado por los datos codificados mediante uno basado en el uno o más elementos de sintaxis.
- 50 10. El procedimiento de la reivindicación 1, en el que la codificación comprende codificar, y en el que la codificación del bloque de datos de vídeo comprende:
 - 55 determinar la paleta sin exceder un número de entradas indicadas por los datos que indican el tamaño máximo de la paleta utilizando el predictor de paleta que no excede el tamaño máximo del predictor de paleta;
 - determinar los valores de índice respectivos para una o más muestras del bloque, en el que los respectivos valores de índice identifican una entrada de la paleta; y
 - 60 codificar los valores de índice en el flujo de bits.
11. El procedimiento de la reivindicación 1, en el que la codificación comprende decodificar, y en el que la decodificación del bloque de datos de vídeo comprende:

determinar la paleta sin exceder un número de entradas indicadas por los datos que indican el tamaño máximo de la paleta utilizando el predictor de paleta que no excede el tamaño máximo del predictor de paleta;

5 obtener, a partir del flujo de bits, respectivos valores de índice para una o más muestras del bloque, en el que los respectivos valores de índice identifican una entrada de la paleta; y

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

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

medios para codificar al menos uno de los datos (20, 30), dentro de un conjunto de parámetros de secuencia (SPS) de un flujo de bits que contiene el bloque de datos de vídeo (180), que indica un tamaño máximo de paleta de una paleta de valores de color (184) para codificar un bloque de datos de vídeo o datos que indican el tamaño máximo del predictor de paleta de un predictor de paleta para determinar la paleta de valores de color; y

20 medios para codificar el bloque de datos de vídeo de acuerdo con los datos.

13. Un medio no transitorio legible por ordenador que tiene instrucciones almacenadas en el mismo que, al ejecutarse, hacen que uno o más procesadores:

25 codifiquen al menos uno de los datos, dentro de un conjunto de parámetros de secuencia (SPS) de un flujo de bits que contiene el bloque de datos de vídeo (180), que indica un tamaño máximo de paleta de una paleta de valores de color (184) para codificar un bloque de datos de vídeo o datos que indican el tamaño máximo del predictor de paleta de un predictor de paleta para determinar la paleta de valores de color; y

30 codifiquen el bloque de datos de vídeo de acuerdo con los datos.

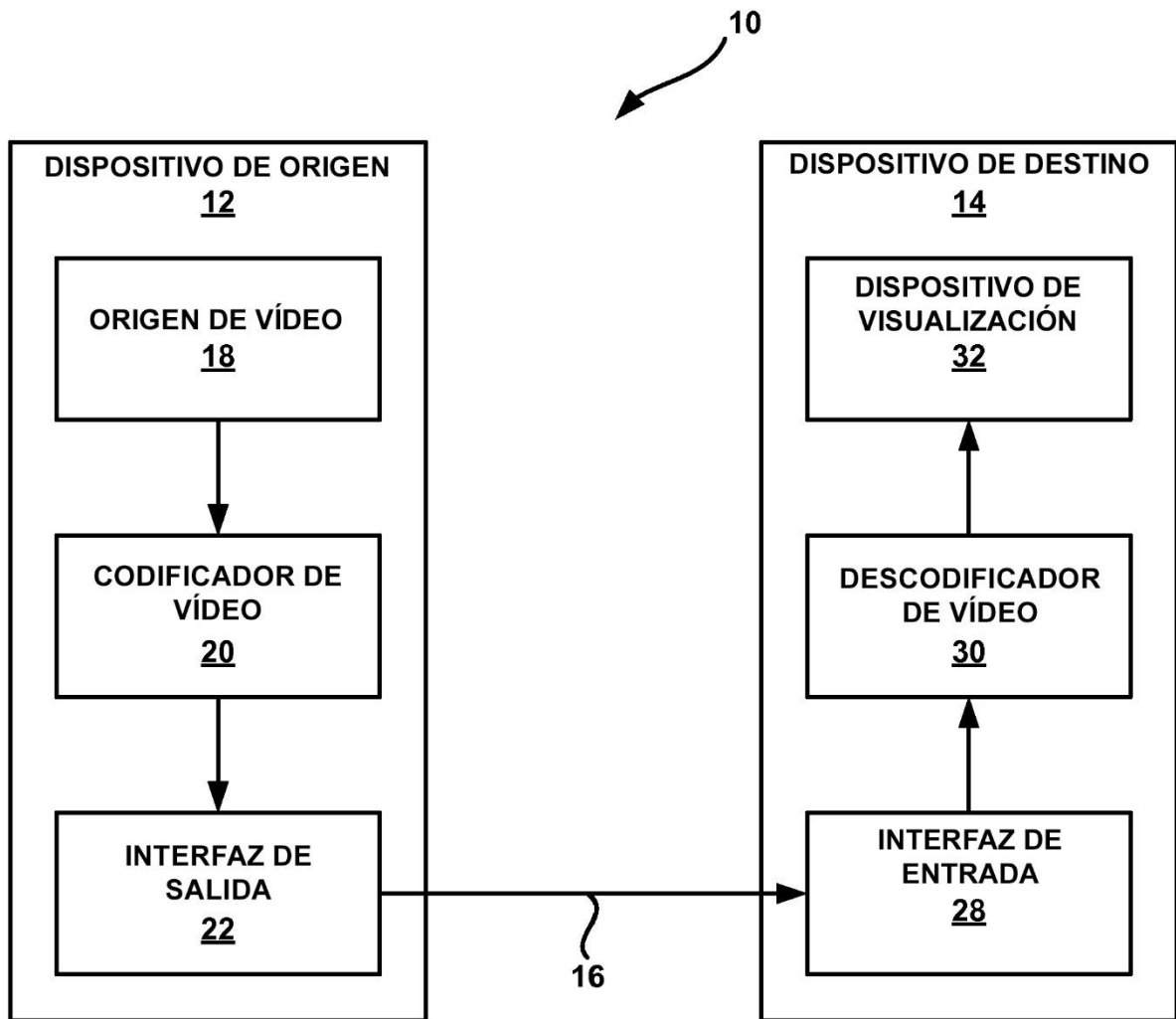


FIG. 1

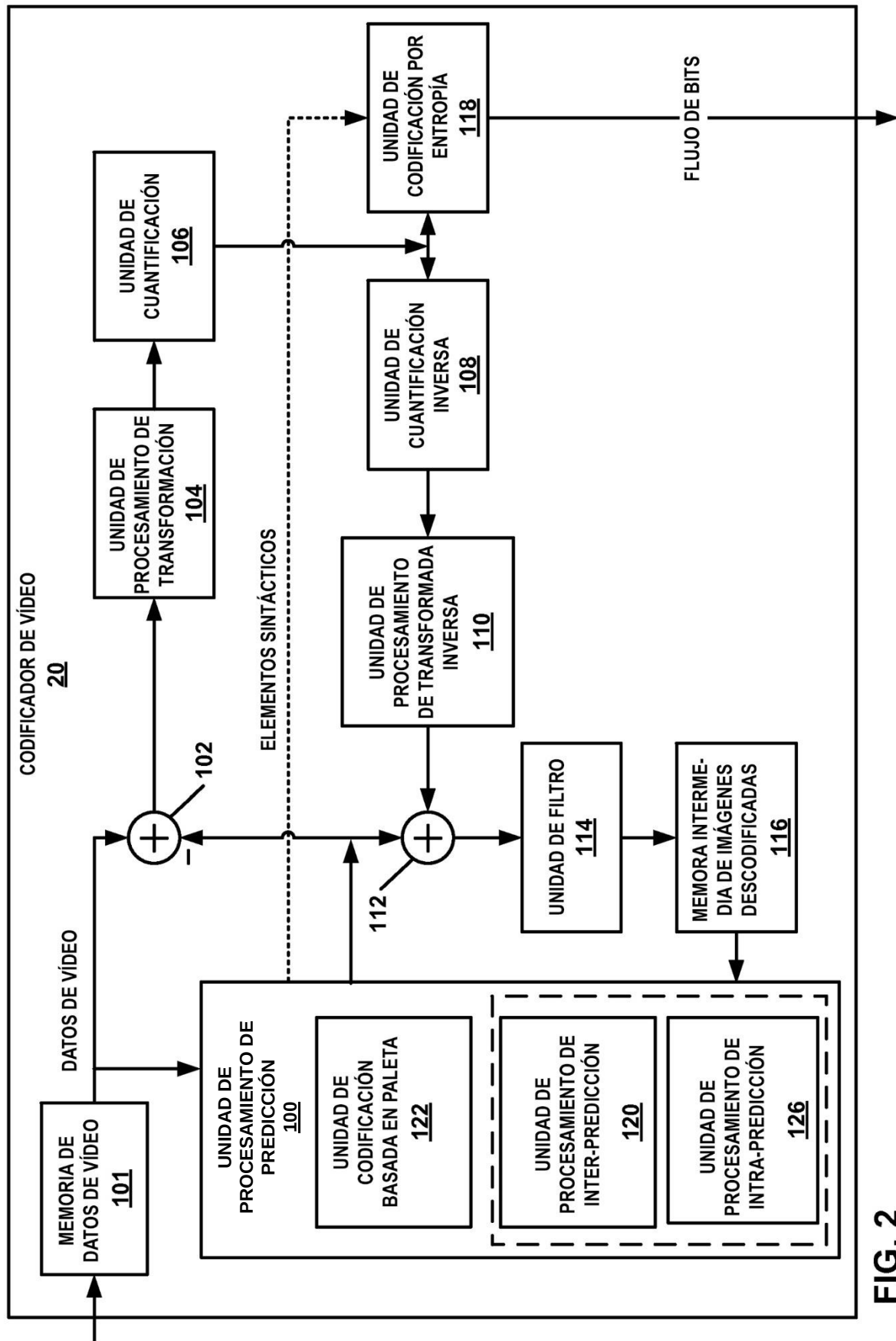


FIG. 2

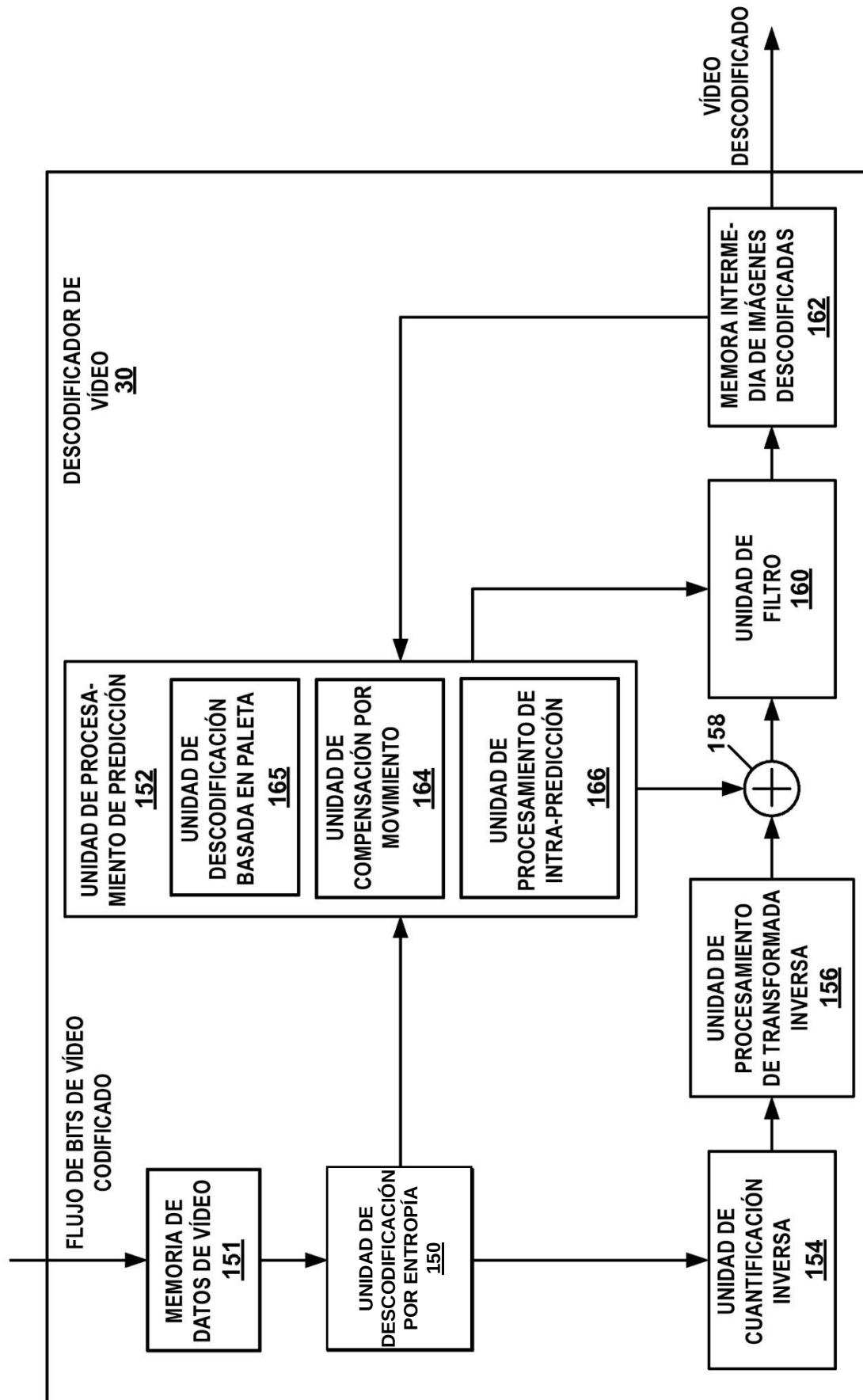


FIG. 3

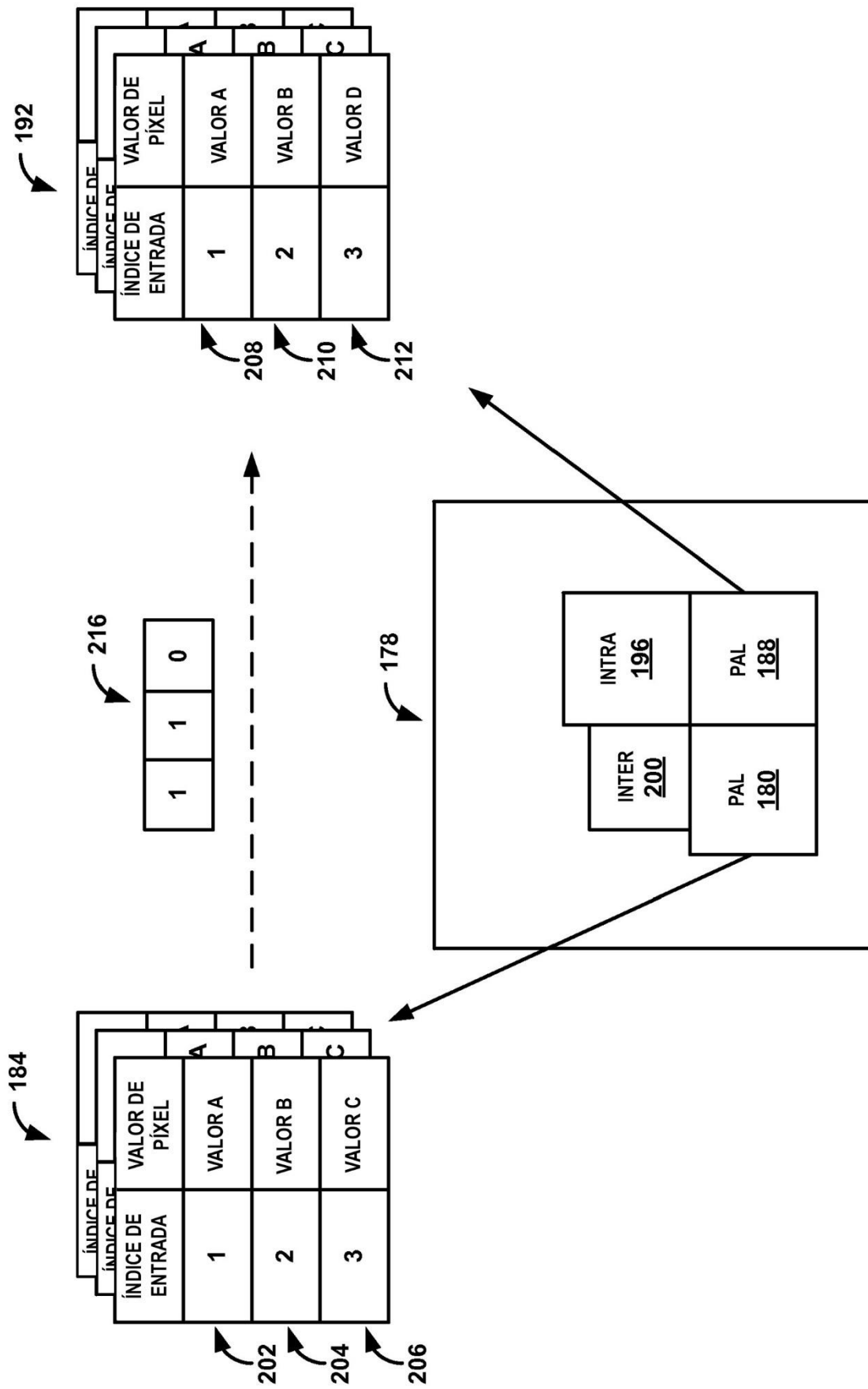


FIG. 4

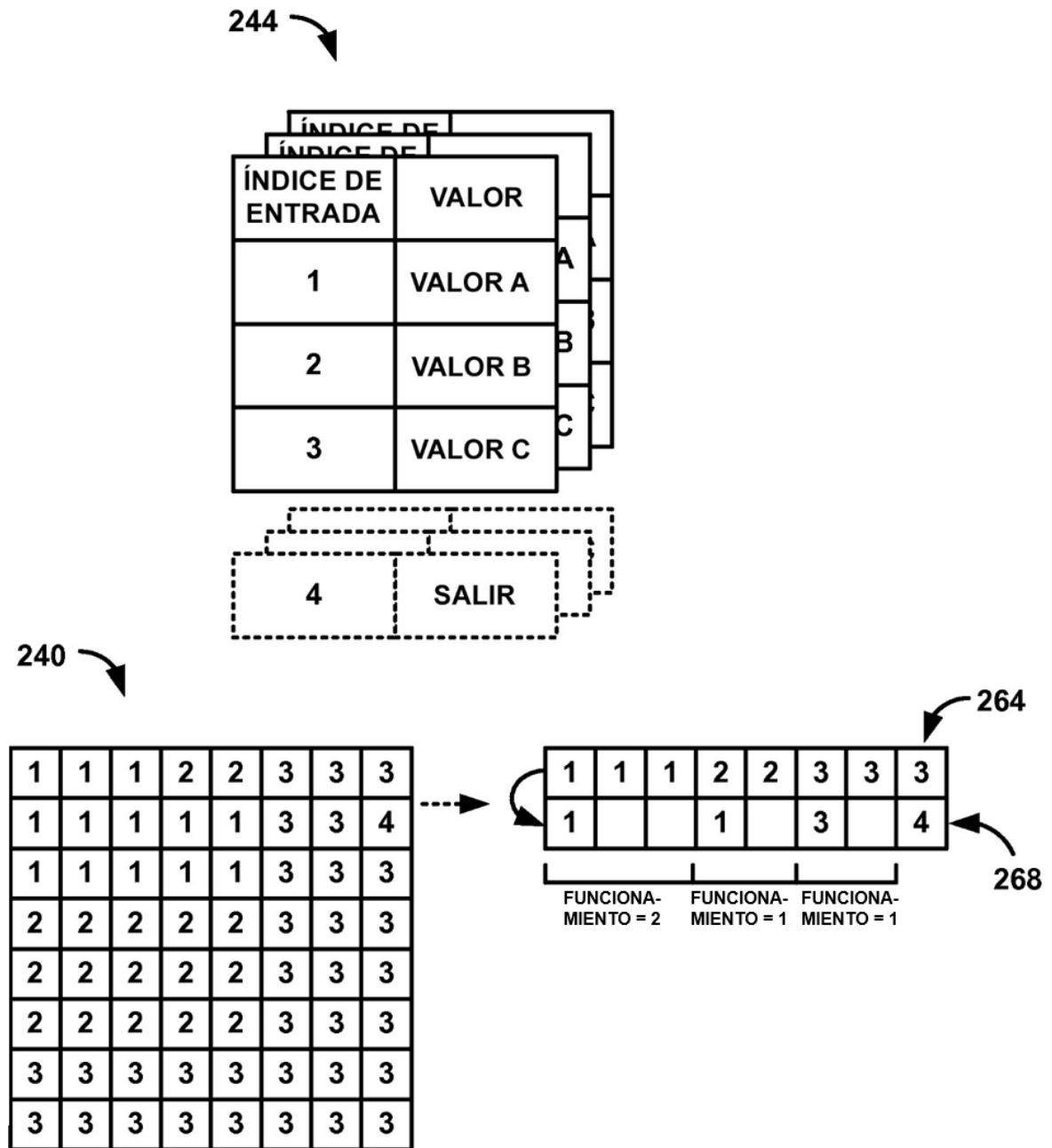


FIG. 5

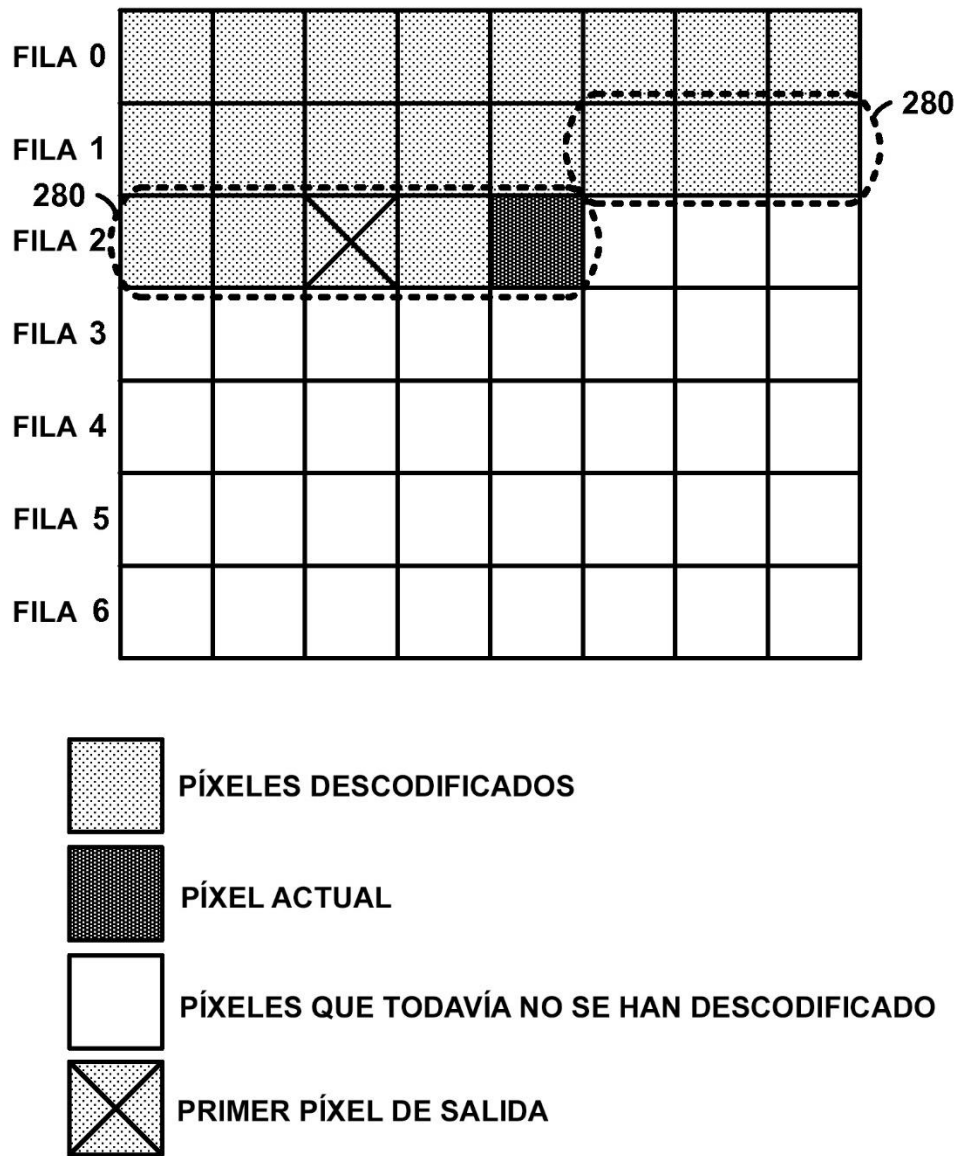


FIG. 6

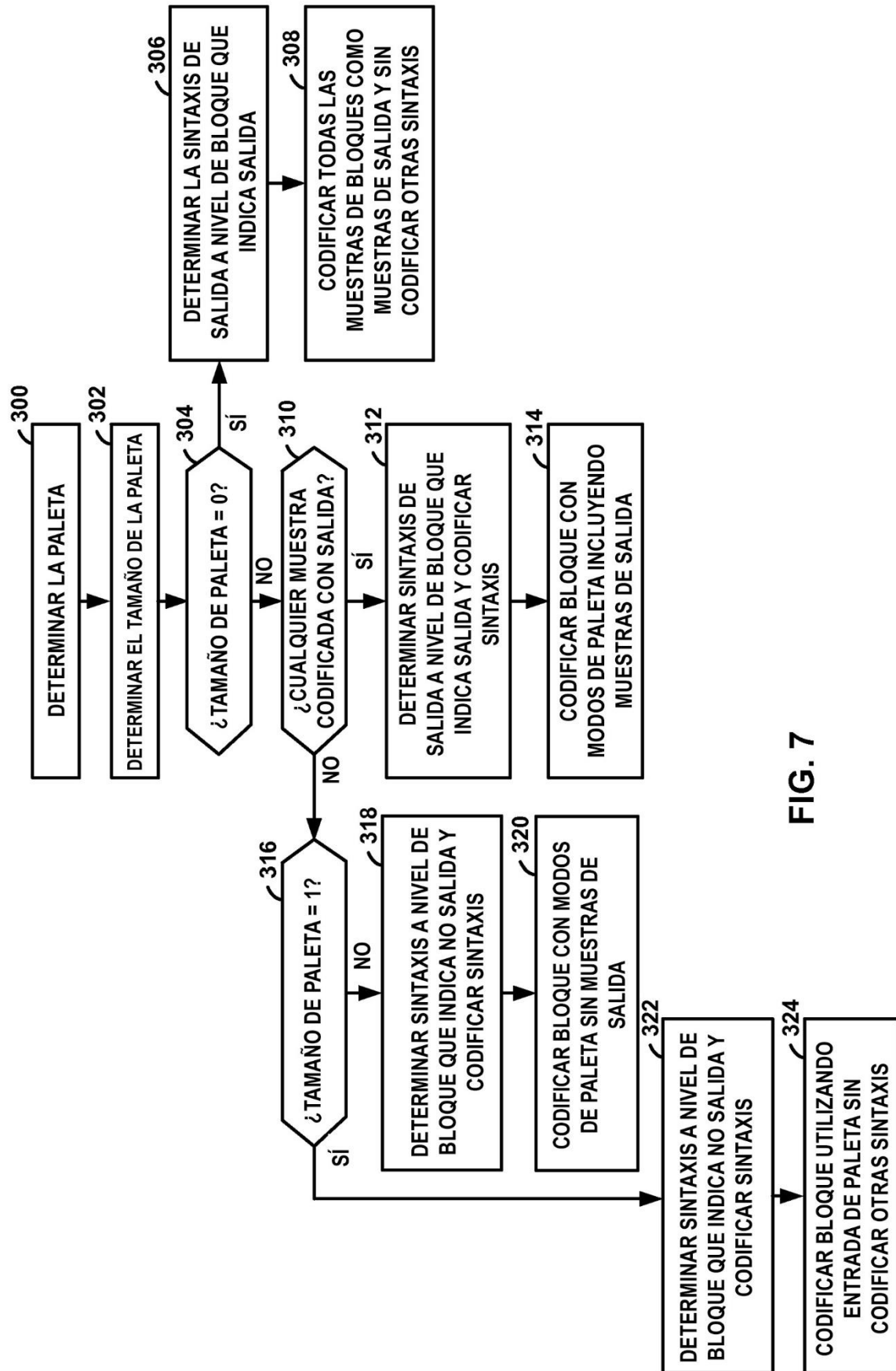


FIG. 7

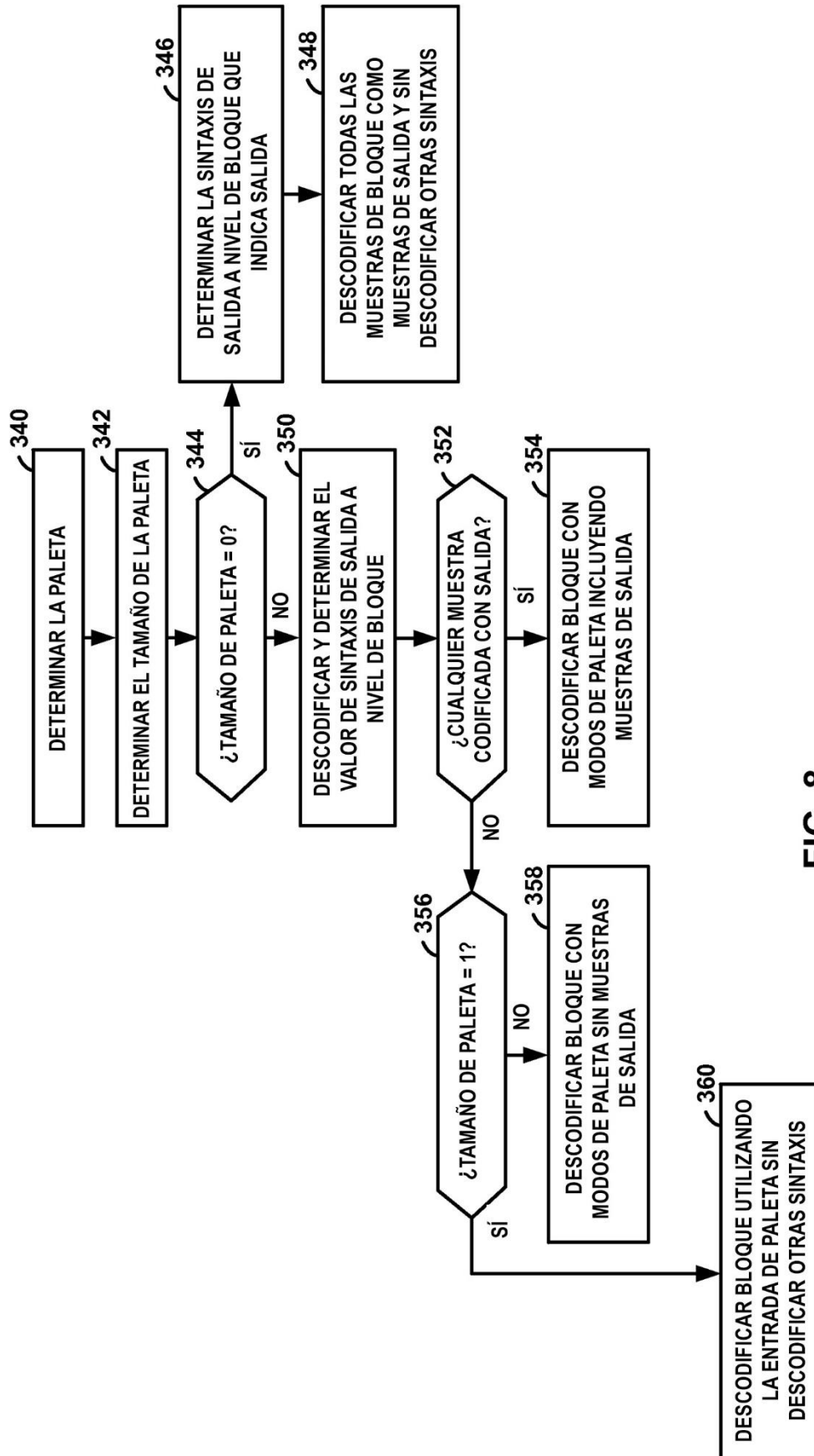


FIG. 8

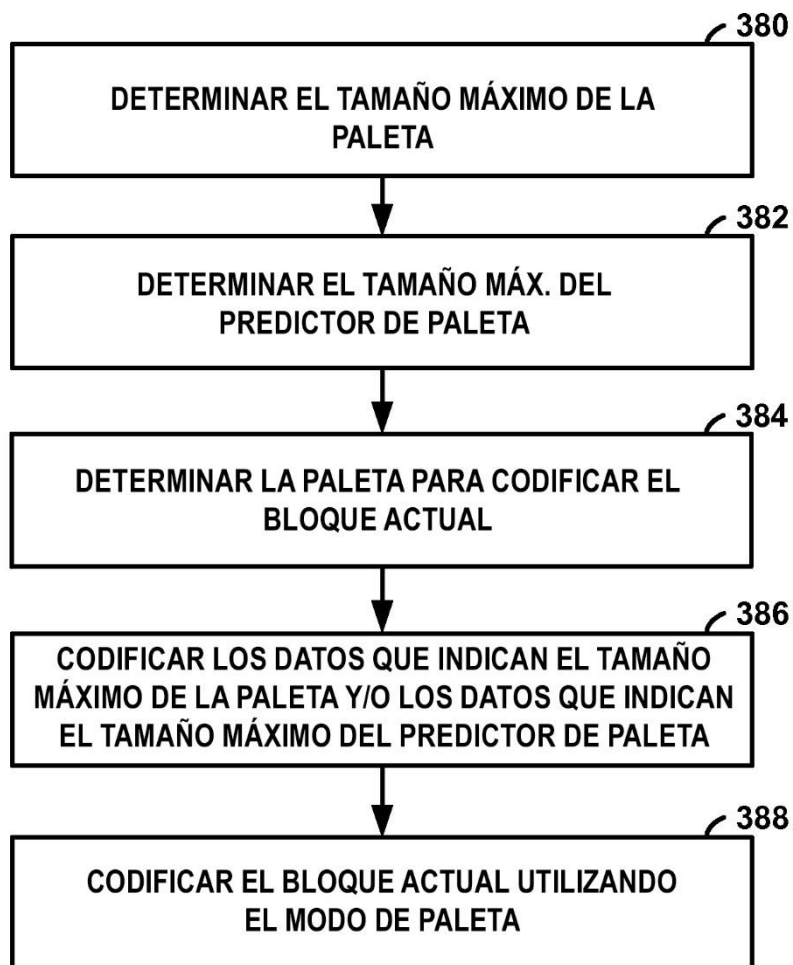


FIG. 9

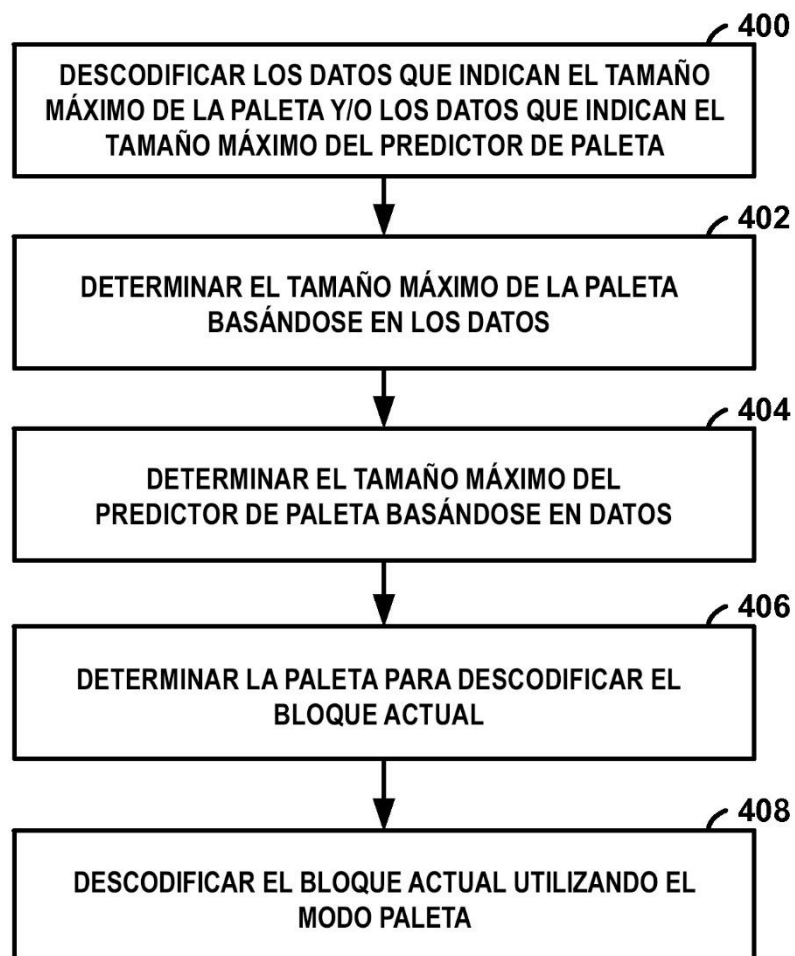


FIG. 10

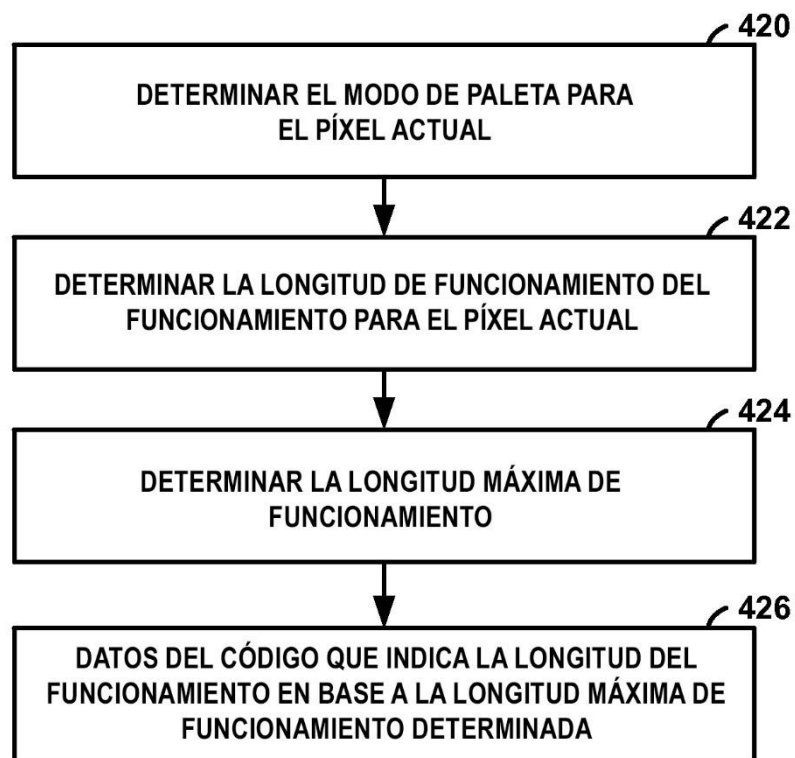


FIG. 11