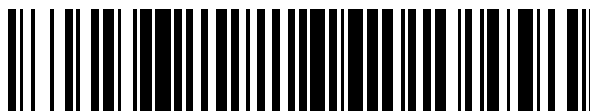


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 702 909**

51 Int. Cl.:

H04N 19/593 (2014.01)
H04N 19/94 (2014.01)
H04N 19/124 (2014.01)
H04N 19/18 (2014.01)
H04N 19/46 (2014.01)
H04N 19/176 (2014.01)
H04N 19/127 (2014.01)
H04N 19/14 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **31.12.2014 PCT/US2014/072999**
- 87 Fecha y número de publicación internacional: **09.07.2015 WO15103392**
- 96 Fecha de presentación y número de la solicitud europea: **31.12.2014 E 14828648 (7)**
- 97 Fecha y número de publicación de la concesión europea: **26.09.2018 EP 3090552**

54 Título: **Cuantización de los píxeles de escape de un bloque de vídeo en modo de codificación de paleta**

30 Prioridad:

02.01.2014 US 201461923163 P
06.01.2014 US 201461924141 P
24.03.2014 US 201461969759 P
30.12.2014 US 201414586820

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
06.03.2019

73 Titular/es:

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

72 Inventor/es:

PU, WEI;
KARCZEWICZ, MARTA;
SOLE ROJALS, JOEL;
GUO, LIWEI;
JOSHI, RAJAN LAXMAN y
ZOU, FENG

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 702 909 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

Cuantización de los píxeles de escape de un bloque de vídeo en modo de codificación de paleta

5 **[0001]** Esta solicitud reivindica el beneficio de:

la Solicitud de Patente Provisional estadounidense No. 61/923,163, presentada el 2 de enero de 2014;

10 la Solicitud de Patente Provisional estadounidense No. 61/924,141, presentada el 6 de enero de 2014; y

la Solicitud Provisional estadounidense No. 61/969,759, presentada el 24 de marzo de 2014.

CAMPO TÉCNICO

15 **[0002]** Esta divulgación se refiere a la codificación y a la decodificación de vídeo.

ANTECEDENTES

20 **[0003]** Las capacidades de 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 móviles o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua 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 de vídeo avanzada (AVC), la norma de codificación de vídeo de alta eficiencia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, decodificar y/o almacenar información de vídeo digital con más eficiencia implementando dichas técnicas de compresión de vídeo.

30 **[0004]** Las técnicas de compresión de vídeo realizan la predicción espacial (intraimagen) y/o la predicción temporal (interimagen) para reducir o eliminar la redundancia inherente a las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (es decir, una trama o una porción de vídeo de una trama de vídeo) puede partitionarse en bloques de vídeo. Los bloques de vídeo de un fragmento intracodificado (I) de una imagen se codifican mediante predicción espacial con respecto a unas muestras de referencia de unos bloques adyacentes en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas, y las imágenes de referencia pueden denominarse tramas de referencia.

40 **[0005]** La predicción espacial o temporal da como resultado un bloque predictivo para un bloque que se vaya a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original que se vaya a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo con un vector de movimiento que apunta a un bloque de muestras de referencia que forman el bloque predictivo, y los datos residuales indican la diferencia entre el bloque codificado y el bloque predictivo. Un bloque intracodificado se codifica de acuerdo con un modo de intracodificación y los datos residuales. Para una mayor compresión, los datos residuales pueden transformarse desde el dominio de píxel en un dominio de transformada, dando como resultado coeficientes residuales, que pueden cuantizarse entonces. Los coeficientes cuantizados, inicialmente dispuestos en una matriz bidimensional, pueden escanearse con el fin de producir un vector unidimensional de coeficientes, y puede aplicarse la codificación por entropía para lograr incluso más compresión.

50 **[0006]** Un flujo de bits de codificación multivistas puede generarse codificando vistas, por ejemplo, desde múltiples perspectivas. Se han desarrollado algunas normas de vídeo tridimensionales (3D) que hacen uso de aspectos de codificación multivistas. Por ejemplo, diferentes visualizaciones pueden transmitir visualizaciones de ojo izquierdo y derecho para soportar vídeo 3D. De forma alternativa, algunos procesos de codificación de vídeo 3D pueden aplicar la denominada codificación multivistas más profundidad. En la codificación multivistas más profundidad, un flujo de bits de vídeo 3D puede contener no solo componentes de visualización de textura, sino también componentes de visualización de profundidad. Por ejemplo, cada visualización puede comprender un componente de visualización de textura y un componente de visualización de profundidad.

60 **[0007]** ZHAOTAI PAN ET AL: "A Low-Complexity Screen Compression Scheme for Interactive Screen Sharing" ["Un esquema de compresión de pantalla de baja complejidad para compartir pantallas interactivas"], TRANSACCIONES DE IEEE EN CIRCUITOS Y SISTEMAS PARA TECNOLOGÍA DE VÍDEO, CENTRO DE SERVICIO IEEE, PISCATAWAY, NJ, EE. UU., Vol. 23, no. 6, 1 de junio de 2013 (2013-06-01), páginas 949-960, XP011513107, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2013.2243056 divulga un algoritmo para comprimir bloques textuales usando colores base y colores de escape para representar y cuantizar los píxeles textuales.

5 **[0008]** GUO X ET AL: "AHG8: Major-color-based screen content coding" ["AHG8: Codificación del contenido de la pantalla basada en el color principal"], 15. REUNIÓN JCT-VC; 23-10-2013 - 1-11-2013; GINEBRA; (EQUIPO COLABORATIVO CONJUNTO SOBRE LA CODIFICACIÓN DE VIDEO DE ISO/CEI JTC1/SC29/WG11 Y UIT-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, no. JCTVC-00182, 14 de octubre de 2013 (2013-10-14), XP030115218, propone un procedimiento basado en el color principal para la codificación del contenido de la pantalla.

10 **[0009]** FLYNN D ET AL: "BoG report on Range Extensions topics" ["Informe BoG sobre temas de extensiones de rango"], 15. REUNIÓN JCT-VC; 23-10-2013 - 1-11-2013; GINEBRA; (EQUIPO COLABORATIVO CONJUNTO SOBRE LA CODIFICACIÓN DE VIDEO DE ISO/CEI JTC1/SC29/WG11 Y UIT-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/, no. JCTVC-00352-v5, 1 de noviembre de 2013 (2013-11-01), XP030115433, es un informe que contiene en particular en la página 37 un comentario sobre el procedimiento de GUO X ET AL.

15 **RESUMEN**

[0010] La invención se define en las reivindicaciones adjuntas.

20 **[0011]** En general, las técnicas de esta divulgación se relacionan con la codificación de vídeo basada en paletas. En la codificación basada en paletas, un codificador de vídeo (por ejemplo, un codificador de vídeo o un decodificador de vídeo) puede formar una denominada "paleta" como una tabla de colores o valores de píxeles que representan los datos de vídeo de un área particular (por ejemplo, un bloque dado). De esta forma, en lugar de codificar los valores de píxeles reales o sus residuales para un bloque actual de datos de vídeo, el codificador de vídeo puede codificar valores de índice de color o paleta para uno o más de los valores de píxeles del bloque actual, donde los valores de índice indican entradas en la paleta que se usan para representar los valores de píxeles del bloque actual. Un mapa de valores de índice de paleta para un bloque actual de datos de vídeo se puede codificar línea por línea usando un orden de escaneo dado y técnicas de codificación de longitud de ejecución. Cada uno de los valores de índice en una línea dada del mapa puede codificarse explícitamente, predecirse a partir de un índice de modo izquierdo de la línea dada, o predecirse a partir de un índice colocado en una línea por encima de la línea dada.

30 **[0012]** Diversas técnicas de esta divulgación están dirigidas a mejorar las técnicas de codificación basada en paletas existentes. En algunos aspectos, esta divulgación está dirigida a técnicas para evitar la codificación de un mapa de valores de índice de paleta para un bloque, si el bloque cumple con ciertos criterios. En algunos aspectos, esta divulgación está dirigida a determinar un rango máximo de valores (también denominado en el presente documento "límite de error") para una paleta determinada usando una tabla de asignación que almacena una relación entre los valores de los parámetros de cuantización y los límites de error de paleta. En algunos aspectos, esta divulgación se dirige a la definición de un parámetro de cuantización para los píxeles de un bloque codificado por paleta que no se asignan a una entrada en la paleta correspondiente (denominados en el presente documento "píxeles de escape") basándose en los parámetros de cuantización usados para la codificación de coeficiente tradicional en un canal de color correspondiente.

45 **[0013]** En un ejemplo, esta divulgación se dirige a un procedimiento de decodificación de datos de vídeo, que incluye determinar un número de entradas incluidas en una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a decodificar, y determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no está asociado con ninguna de las entradas en la paleta. El procedimiento puede incluir además una respuesta a la determinación de que el número de entradas incluidas en la paleta es igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, omitiendo la decodificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo, y a la determinación de los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta.

55 **[0014]** En otro ejemplo, esta divulgación está dirigida a un procedimiento de codificación de datos de vídeo, incluyendo el procedimiento determinar un número de entradas incluidas en una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, y determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no esté asociado con ninguna de las entradas en la paleta. El procedimiento puede incluir además una respuesta a la determinación de que el número de entradas incluidas en la paleta es igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, omitiendo la codificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo y la codificación del bloque de datos de vídeo mediante la determinación de los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta.

65 **[0015]** En otro ejemplo, esta divulgación se dirige a un aparato para decodificar datos de vídeo, comprendiendo el aparato una memoria configurada para almacenar datos de vídeo y uno o más procesadores configurados para determinar varias entradas incluidas en una paleta usada para representar valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, y para determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no esté asociado con ninguna de las entradas en la paleta. El uno o más procesadores también pueden configurarse para responder a una determinación de que el número de entradas incluidas en la paleta es igual a uno

y que el bloque de datos de vídeo no incluye al menos un píxel de escape, omitiendo la codificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo, y para determinar los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta.

5 **[0016]** En otro ejemplo, esta divulgación se dirige a un aparato para codificar datos de vídeo, comprendiendo medios para determinar un número de entradas incluidas en una paleta usada para representar valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, y medios para determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no esté asociado con ninguna de las entradas en la paleta. El aparato puede incluir además medios para omitir, en respuesta a la determinación de que el número de entradas incluidas en la paleta es
10 igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, la codificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo y los medios para determinar los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta.

15 **[0017]** En otro ejemplo, esta divulgación se dirige a un medio no transitorio legible por ordenador codificado con instrucciones que, cuando se ejecutan, causan que uno o más procesadores de un dispositivo para codificar datos de vídeo determinen una cantidad de entradas incluidas en una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, y que determinen si el bloque de datos de vídeo incluye al menos un píxel de escape que no esté asociado con ninguna de las entradas en la paleta. Las instrucciones, cuando se ejecutan, pueden causar que uno o más procesadores respondan a una determinación de que el número de entradas
20 incluidas en la paleta es igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, omitiendo la codificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo, y para determinar los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta.

25 **[0018]** En un ejemplo, esta divulgación está dirigida a un procedimiento de decodificación de datos de vídeo, que incluye la determinación de una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a decodificar, e identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de las entradas en la paleta. El procedimiento puede incluir además identificar un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque para un canal de color
30 dado basándose en un valor de QP usado para la codificación de coeficiente de transformada en la codificación no basada en paletas usando cada uno de los uno o más píxeles de escape el único valor de QP identificado. El procedimiento puede incluir además determinar los valores de píxeles del bloque de datos de vídeo basándose en los píxeles de escape descuantizados y en los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con una o más entradas en la paleta.

35 **[0019]** En otro ejemplo, esta divulgación está dirigida a un procedimiento de codificación de datos de vídeo, incluyendo el procedimiento determinar una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar e identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no están asociados con ninguna de las entradas en la paleta. El procedimiento puede incluir además la identificación de un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de QP usado para la codificación del coeficiente de transformada en la
40 codificación no basada en paletas, y la cuantización cada uno de los uno o más píxeles de escape usando el único valor de QP identificado. El procedimiento puede incluir además la codificación de los valores de píxeles del bloque de datos de vídeo, incluidos los píxeles de escape cuantizados y los valores de índice para uno o más píxeles en el bloque de datos de vídeo que estén asociados con una o más entradas en la paleta.

45 **[0020]** En otro ejemplo, esta divulgación se dirige a un aparato para codificar datos de vídeo, comprendiendo el aparato una memoria configurada para almacenar datos de vídeo y uno o más procesadores en comunicación con la memoria y configurados para determinar una paleta usada para representar valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, e identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de una o más entradas en la paleta. El uno o más procesadores pueden configurarse además para identificar un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de QP usado para la codificación del coeficiente de transformada en la codificación no basada en paletas, y para aplicar el único valor de QP identificado a cada uno de
50 los uno o más píxeles de escape. El uno o más procesadores pueden configurarse además para determinar los valores de píxeles del bloque de datos de vídeo basándose en los píxeles de escape y los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con una o más entradas.

55 **[0021]** En otro ejemplo, esta divulgación se dirige hacia un aparato para codificar datos de vídeo, comprendiendo el aparato medios para determinar una paleta usada para representar valores de píxeles de un bloque de datos de vídeo que se vaya a codificar, medios para identificar, en el bloque datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de las entradas en la paleta, medios para identificar un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque para un canal de color determinado basado en un valor de QP usado para la codificación del coeficiente de transformada en la codificación no basada en paletas,
60 medios para aplicar el valor de QP único identificado a cada uno de los uno o más píxeles de escape, y medios para determinar los valores de píxeles del bloque de datos de vídeo basándose en los píxeles de escape y en los valores

de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con una o más entradas.

[0022] En otro ejemplo, esta divulgación se dirige hacia un medio no transitorio legible por ordenador codificado con instrucciones que, al ejecutarse, causan que uno o más procesadores de un dispositivo para codificar datos de vídeo determinen una paleta usada para representar los valores de píxeles de un bloque de los datos de vídeo que se vayan a codificar, que identifiquen, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de las entradas en la paleta, y que identifiquen un único valor de parámetro de cuantización (QP) para todos del uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de QP usado para la codificación del coeficiente de transformada en la codificación no basada en paletas. Las instrucciones, cuando se ejecutan, pueden causar que uno o más procesadores apliquen el único valor de QP identificado a cada uno de los uno o más píxeles de escape, y que determinen los valores de píxeles del bloque de datos de vídeo de acuerdo con los píxeles de escape y los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con una o más entradas.

[0023] Las técnicas descritas en el presente documento pueden proporcionar una o más ventajas y mejoras potenciales sobre las técnicas de codificación basada en paletas existentes y/o las técnicas de compresión de datos. Por ejemplo, diversas técnicas de esta divulgación pueden implementarse mediante dispositivos de codificación de vídeo para conservar los recursos informáticos y los requisitos de ancho de banda, mientras se mantiene la precisión de los datos. Además, diversas técnicas de esta divulgación pueden implementarse mediante dispositivos de codificación de vídeo para mejorar la eficiencia y precisión de las técnicas de codificación basada en paletas existentes y las técnicas de compresión de datos.

[0024] 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

[0025]

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

La FIG. 2 es un diagrama 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 de flujo que ilustra un proceso de ejemplo mediante el cual un dispositivo de decodificación de vídeo puede implementar técnicas de esta divulgación para evitar la decodificación de valores de índice para píxeles de un bloque codificado por paleta, basándose en un conjunto particular de condiciones.

La FIG. 5 es un diagrama de flujo, un proceso de ejemplo mediante el cual un dispositivo de codificación de vídeo puede implementar técnicas de esta divulgación para omitir la codificación de valores de índice para píxeles de un bloque codificado por paleta, basado en un conjunto particular de condiciones.

La FIG. 6 es un diagrama de flujo que ilustra un proceso de ejemplo mediante el cual un dispositivo de decodificación de vídeo puede implementar técnicas de esta divulgación para descuantizar uno o más píxeles de escape de un bloque de datos de vídeo codificado por paleta.

La FIG. 7 es un diagrama de flujo que ilustra un proceso de ejemplo mediante el cual un dispositivo de codificación de vídeo puede implementar técnicas de esta divulgación para cuantizar uno o más píxeles de escape de un bloque de datos de vídeo codificado por paleta.

DESCRIPCIÓN DETALLADA

[0026] Esta divulgación incluye técnicas para codificación y compresión de vídeo. En general, esta divulgación describe técnicas para la codificación basada en paletas de datos de vídeo. En la codificación de vídeo tradicional, se supone que las imágenes son de tono continuo y espacialmente fluidas. Basándose en estas suposiciones, se han desarrollado diversas herramientas, tales como la transformada basada en bloques, el filtrado, etc., y dichas herramientas han demostrado un buen rendimiento para vídeos de contenido natural.

[0027] Sin embargo, en aplicaciones como escritorio remoto, trabajo colaborativo y pantalla inalámbrica, el contenido de la pantalla generado por ordenador (por ejemplo, tales como texto o gráficos de ordenador) puede ser el contenido

dominante que se vaya a comprimir. Este tipo de contenido tiende a tener tonos discretos, y presenta líneas nítidas y límites de objetos de alto contraste. Es posible que la suposición de tono continuo y suavidad ya no se aplique al contenido de la pantalla, y por tanto que las técnicas tradicionales de codificación de vídeo puedan no ser formas eficientes de comprimir los datos de vídeo, incluido el contenido de la pantalla.

5
 [0028] Esta divulgación describe la codificación basada en paletas, que puede ser particularmente adecuada para la codificación de contenido generado por la pantalla. Por ejemplo, asumiendo 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 decodificador de vídeo) puede formar una llamada "paleta" para representar los datos de vídeo del área particular. La paleta puede expresarse como una tabla de colores o valores de píxeles que representan los datos de vídeo del área en particular (por ejemplo, un bloque dado). Por ejemplo, la paleta puede incluir los valores de píxeles más dominantes en el bloque dado. En algunos casos, los valores de píxeles más dominantes pueden incluir uno o más valores de píxeles que se produzcan con mayor frecuencia dentro del bloque. Además, en algunos casos, un codificador de vídeo puede aplicar un valor umbral para determinar si un valor de píxel se incluirá como uno de los valores de píxeles más dominantes en el bloque. De acuerdo con diversos aspectos de la codificación basada en paletas, el codificador de vídeo puede codificar valores de índice indicativos de uno o más de los valores de píxeles del bloque actual, en lugar de codificar valores de píxeles reales o sus residuos para un bloque actual de datos de vídeo. En el contexto de la codificación basada en paletas, los valores de índice indican entradas respectivas en la paleta que se usan para representar valores de píxeles individuales del bloque actual.

20
 [0029] Por ejemplo, el codificador de vídeo puede codificar un bloque de datos de vídeo determinando la paleta para el bloque (por ejemplo, codificando la paleta explícitamente, prediciendo la paleta, o una combinación de los mismos), ubicando una entrada en la paleta para representar uno o más de los valores de píxeles, y la codificación del bloque con valores de índice que indican la entrada en la paleta utilizada para representar los valores de píxeles del bloque. En algunos ejemplos, el codificador de vídeo puede señalar la paleta y/o los valores de índice en un flujo de bits codificado. A su vez, el decodificador de vídeo puede obtener, a partir de un flujo de bits codificado, una paleta para un bloque, así como valores de índice para los píxeles individuales del bloque. El decodificador de vídeo puede relacionar los valores de índice de los píxeles con las entradas de la paleta para reconstruir los valores de píxeles del bloque.

30
 [0030] La codificación basada en paletas de datos de vídeo se ha descrito en detalle anteriormente. La idea básica de la codificación basada en paletas es que, para cada CU, se deriva una paleta que comprende (y puede consistir en) los valores de píxeles más dominantes en la CU actual. El tamaño y los elementos de la paleta se transmiten primero desde un codificador de vídeo a un decodificador de vídeo. Después de eso, los valores de píxeles en la CU se codifican de acuerdo con un cierto orden de escaneo. Para cada ubicación de píxeles en la CU, primero se transmite un indicador, por ejemplo, `palette_flag`, para indicar si el valor de píxel está incluido en la paleta (es decir, "modo de ejecución") o no (es decir, "modo de píxel"). En el "modo de ejecución", se señala el índice de paleta asociado con la ubicación del píxel en la CU, seguido de una "ejecución" del valor de píxel. Ni el indicador de paleta ni el índice de paleta deben transmitirse para las siguientes ubicaciones de píxeles que están cubiertas por la "ejecución" ya que todas tienen el mismo valor de píxel. En el "modo de píxel", el valor de píxel se transmite para la ubicación de píxel dada en la CU.

45
 [0031] Para cada CU, se deriva una tabla de color principal que consta de los valores de píxeles más dominantes en la CU actual. El tamaño y los elementos de la tabla se transmiten primero. El tamaño y/o los elementos de la tabla de colores principales pueden codificarse directamente o codificarse de manera predecible usando el tamaño y/o los elementos de la tabla de colores principales en las CU adyacentes (por ejemplo, la CU con codificación superior y/o izquierda).

50
 [0032] En algunos ejemplos, cada uno de los píxeles en la CU actual se asigna a un índice de tabla de colores principales. Para aquellos píxeles cuyos índices de colores principales no existen, se les asigna un índice especial (denominado 'otro índice') y estos píxeles se denominan 'píxel de escape'. Las técnicas de esta divulgación se centran en el procedimiento de codificación de los índices de colores principales.

55
 [0033] Un 'píxel de escape' puede codificarse usando cualquier procedimiento de codificación por entropía existente, tal como la codificación de longitud fija, la codificación unaria, etc. Un procedimiento para codificar los valores de escape es usar una operación de desplazamiento a la izquierda dependiendo del valor del parámetro de cuantización (QP). Es decir, codifique solo los bits más significativos, dependiendo del número de bits del valor de QP. Para ese fin, una estrategia usada en el estado de la técnica es usar una tabla que asigne cada QP a un número que sea el desplazamiento a la derecha que se aplicará al valor del píxel.

60
 [0034] El bloque del índice de color principal se codifica línea por línea. Para cada línea, se elige un modo de línea de 'horizontal', 'vertical' y 'normal'. Si se elige el modo de línea 'horizontal', todos los índices en la línea actual son los mismos que el índice más a la izquierda del índice más a la izquierda en la línea anterior. Si se elige el modo de línea 'vertical', la línea completa se copia de la línea anterior. Si se selecciona el modo 'normal', los índices dentro de la línea actual se codifican uno por uno. Para cada índice en este caso, se usa un elemento sintáctico para indicar si el índice se puede copiar del índice colocado en la línea anterior ('copy_from_top'), o del índice adyacente izquierdo del índice

('copy_from left'), o ninguna copia es posible ('no_copy'). Si no es posible realizar ninguna copia, el píxel se codifica directamente.

[0035] Los ejemplos anteriores pretenden proporcionar una descripción general de la codificación basada en paletas. En diversos ejemplos, las técnicas descritas en esta divulgación pueden incluir técnicas para diversas combinaciones de uno o más de los datos de vídeo codificados de señalización formados por modos de codificación basados en paletas, paletas de transmisión, paletas de predicción, paletas derivadas o decodificación de datos de vídeo de mapas de codificación basada en paletas y otros elementos sintácticos. Dichas técnicas pueden mejorar la eficiencia de la codificación de vídeo, por ejemplo, requerir menos bits para representar el contenido generado por la pantalla.

[0036] Esta divulgación describe diversas técnicas relacionadas con la codificación de vídeo basada en paletas. En algunos aspectos, esta divulgación está dirigida a omitir la codificación de un mapa de valores de índice para un bloque de vídeo cuando una paleta asociada con el bloque de vídeo solo incluye una entrada o color, y el bloque de vídeo no incluye ningún píxel de escape. En algunos aspectos, esta divulgación está dirigida a derivar un "límite de error" de la paleta, que indica una desviación máxima de un valor de píxel fijo que puede incluirse dentro de la paleta correspondiente, utilizando una tabla de asignación que almacena una relación entre los valores de los parámetros de cuantización y los límites de error de paletas. Algunos aspectos de esta divulgación están dirigidos a derivar un parámetro de cuantización (QP) para la cuantización y el recuento de un píxel de escape (o el error de predicción asociado), usando parámetros de cuantización usados para la codificación de coeficientes tradicionales en un canal de color correspondiente. Adicionalmente, esta divulgación describe el uso de un indicador para indicar si un píxel que se incluye en un área codificada basada en paletas es un píxel de escape. Los aspectos de esta divulgación también describen técnicas para cuantizar un valor de píxel de escape, tales como las técnicas que emplean una operación de desplazamiento hacia la derecha.

[0037] En algunos aspectos, esta divulgación se dirige a la cuantización de píxeles de escape de un bloque codificado por paleta de acuerdo con una función de asignación (por ejemplo, una función de desplazamiento) basada en un valor de parámetro de cuantización para el píxel de escape. En algunos aspectos, esta divulgación está dirigida a usar un indicador para indicar y/o determinar si un píxel de un bloque codificado por paleta es un píxel de escape.

[0038] En algunos ejemplos de esta divulgación, las técnicas para la codificación basada en paletas de datos de vídeo se pueden usar con una o más técnicas de codificación adicionales, tales como técnicas para la codificación interpredictiva o la codificación intrapredictiva de datos de vídeo. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o decodificador, o un codificador-decodificador combinado (códec), se puede configurar para realizar una codificación interpredictiva e intrapredictiva, así como una codificación basada en paletas. En algunos ejemplos, las técnicas de codificación basada en paletas pueden configurarse para su uso en uno o más modos de codificación de Codificación de Vídeo de Alta Eficiencia (HEVC). En otros ejemplos, las técnicas de codificación basada en paletas se pueden usar de forma independiente o como parte de otros sistemas o normas existentes o futuros.

[0039] La Codificación de Vídeo de Alta Eficiencia (HEVC) es una nueva norma de codificación de vídeo desarrollada por el Equipo Conjunto de Colaboración en Codificación de Vídeo (JCT-VC) del Grupo de Expertos en Codificación de Vídeo del UIT-T (VCEG) y el Grupo de Expertos en Imagen en Movimiento ISO/IEC (MPEG). Un borrador reciente de la norma HEVC, denominado "Borrador 10 de la HEVC" o "WD10", se describe en el documento JCTVC-H1003v34, de Bross et al., "Codificación de Vídeo de Alta Eficiencia (HEVC) Borrador 10 de la Memoria descriptiva (for FDIS & Last Call)" Equipo de Colaboración Conjunta en Codificación de Vídeo (JCT-VC) de ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11, décimo segunda conferencia: Ginebra, CH, 14-23 de enero de 2013, disponible en: http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip. El documento estándar finalizado de HEVC se publica como "UIT-T H.265, SERIE H: SISTEMAS AUDIOVISUALES Y MULTIMEDIA Infraestructura de servicios audiovisuales - Codificación de vídeo en movimiento - Codificación de vídeo de alta eficiencia, "Sector de Normalización de las Telecomunicaciones de la Unión Internacional de Telecomunicaciones (UIT), abril de 2013.

[0040] Con respecto al marco de la HEVC, como ejemplo, las técnicas de codificación basada en paletas se pueden configurar para usarse como un modo de unidad de codificación (CU). En otros ejemplos, las técnicas de codificación basada en paletas pueden configurarse para usarse como un modo de unidad de predicción (PU) en el marco de la HEVC. Por consiguiente, todos los procesos divulgados a continuación, descritos en el contexto de un modo CU, pueden, adicional o alternativamente, aplicarse a PU. Sin embargo, estos ejemplos basados en la HEVC no deberían considerarse como una restricción o limitación de las técnicas de codificación basada en paletas, descritas en el presente documento, ya que dichas técnicas se pueden aplicar para que funcionen independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación por paletas pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[0041] La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación de vídeo 10 de ejemplo que puede utilizar las técnicas de esta divulgación. Como se usa en el presente documento, el término "codificador de vídeo" se refiere genéricamente tanto a codificadores de vídeo como a decodificadores de vídeo. En esta divulgación, los términos "codificación de vídeo" o "codificación" pueden referirse genéricamente a la codificación de vídeo o a la decodificación de vídeo. El codificador de vídeo 20 y el decodificador de vídeo 30 del sistema de codificación de vídeo

10 representan ejemplos de dispositivos que pueden configurarse para realizar técnicas de codificación de vídeo basada en paletas de acuerdo con diversos ejemplos descritos en esta divulgación. Por ejemplo, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar configurados para codificar selectivamente varios bloques de datos de vídeo, tales como las CU o PU en la codificación HEVC, usando la codificación basada en paleta o bien la codificación no basada en paletas. Las modalidades de codificación no basada en paletas pueden referirse a diversas modalidades de codificación temporal interpredictiva o a modalidades de codificación espacial intrapredictiva, tales como las diversas modalidades de codificación especificadas por el Borrador 10 de la HEVC.

[0042] Como se muestra en la FIG. 1, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12 y un dispositivo de destino 14. El dispositivo de origen 12 genera datos de vídeo codificados. Por consiguiente, el dispositivo de origen 12 puede denominarse dispositivo de codificación de vídeo o aparato de codificación de vídeo. El dispositivo de destino 14 puede decodificar datos de vídeo codificados, generados por el dispositivo de origen 12. Por consiguiente, el dispositivo de destino 14 puede denominarse dispositivo de decodificación de vídeo o aparato de decodificación de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 pueden ser ejemplos de dispositivos de codificación de vídeo o aparatos de codificación de vídeo.

[0043] El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender una amplia gama de dispositivos, incluyendo ordenadores de sobremesa, dispositivos informáticos móviles, ordenadores plegables (es decir, portátiles), ordenadores de tableta, decodificadores, equipos telefónicos manuales tales como los denominados teléfonos "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, ordenadores de coche o similares.

[0044] El dispositivo de destino 14 puede recibir datos de vídeo codificados desde el dispositivo de origen 12 a través de un canal 16. El canal 16 puede comprender uno o más medios o dispositivos capaces de mover los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el canal 16 puede comprender uno o más medios de comunicación que permitan al dispositivo de origen 12 transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. En este ejemplo, el dispositivo de origen 12 puede modular los datos de vídeo codificados de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y puede transmitir los datos de vídeo modulados al dispositivo de destino 14. Los uno o más medios de comunicación pueden incluir medios de comunicación inalámbricos y/o alámbricos, tales como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. Los uno o más medios de comunicación pueden formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global (por ejemplo, Internet). Los uno o más medios de comunicación pueden incluir routers, conmutadores, estaciones base u otros equipos que faciliten la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

[0045] En otro ejemplo, el canal 16 puede incluir un medio de almacenamiento que almacene datos de vídeo codificados, generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder al medio de almacenamiento a través del acceso al disco o del acceso a la tarjeta. El medio de almacenamiento puede incluir varios medios de almacenamiento de datos de acceso local, tales como discos Blu-ray, DVD, CD-ROM, memoria flash u otros medios adecuados de almacenamiento digital para almacenar datos de vídeo codificados.

[0046] En un ejemplo adicional, el canal 16 puede incluir un servidor de ficheros u otro dispositivo de almacenamiento intermedio que almacene los datos de vídeo codificados, generados por el dispositivo de origen 12. En este ejemplo, el dispositivo de destino 14 puede acceder a datos de vídeo codificados almacenados en el servidor de ficheros o en otro dispositivo de almacenamiento intermedio mediante transmisión continua o descarga. El servidor de ficheros puede ser un tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir los datos de vídeo codificados al dispositivo de destino 14. Entre los ejemplos de servidores de ficheros se incluyen servidores de red (por ejemplo, para una página web), servidores del protocolo de transferencia de ficheros (FTP), dispositivos de almacenamiento conectados a red (NAS) y unidades de disco local.

[0047] El dispositivo de destino 14 puede acceder a los datos de vídeo codificados a través de una conexión de datos estándar, tal como una conexión a Internet. Entre los ejemplos de tipos de conexiones de datos pueden incluirse canales inalámbricos (por ejemplo, conexiones WiFi), conexiones cableadas (por ejemplo, DSL, módem de cable, etc.) o combinaciones de ambos que sean adecuadas para acceder a datos de vídeo codificados almacenados en un servidor de ficheros. La transmisión de datos de vídeo codificados desde el servidor de ficheros puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0048] Las técnicas de esta divulgación no están limitadas a aplicaciones o a configuraciones inalámbricas. Las técnicas pueden aplicarse a la codificación de vídeo en soporte de varias 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 la transmisión de vídeo unidireccional o bidireccional para soportar aplicaciones tales como la transmisión de vídeo, la reproducción de vídeo, la radiodifusión de vídeo y/o la videotelefonía.

5 [0049] El sistema de codificación de vídeo 10 ilustrado en la FIG. 1 es simplemente un ejemplo y las técnicas de esta divulgación pueden aplicarse a configuraciones de codificación de vídeo (por ejemplo, codificación de vídeo o decodificación de vídeo) que no incluyan necesariamente ninguna comunicación de datos entre los dispositivos de codificación y decodificación. En otros ejemplos, los datos se recuperan de una memoria local, se 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 desde la memoria. En muchos ejemplos, la codificación y la decodificación se realiza mediante dispositivos que no se comunican entre sí, sino que simplemente codifican datos en la memoria y/o recuperan y decodifican datos desde la memoria.

10 [0050] 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/demodulador (módem) y/o un transmisor. La fuente de vídeo 18 puede incluir un dispositivo de captura de vídeo, por ejemplo, una videocámara, un fichero de vídeo que contenga datos de vídeo previamente capturados, una interfaz de alimentación de vídeo para recibir datos de vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos por ordenador para generar datos de vídeo, o una combinación de dichas fuentes de datos de vídeo.

15 [0051] El codificador de vídeo 20 puede codificar datos de vídeo procedentes de la fuente de vídeo 18. En algunos ejemplos, el dispositivo de origen 12 transmite directamente los datos de vídeo codificados al dispositivo de destino 14 mediante la interfaz de salida 22. En otros ejemplos, los datos de vídeo codificados también pueden almacenarse en un medio de almacenamiento o en un servidor de ficheros para un acceso posterior mediante el dispositivo de destino 14 para su decodificación y/o su reproducción.

20 [0052] En el ejemplo de la FIG. 1, el dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos ejemplos, la interfaz de entrada 28 incluye un receptor y/o un módem. La interfaz de entrada 28 puede recibir los datos de vídeo codificados por el canal 16. El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En general, el dispositivo de visualización 32 muestra los datos de vídeo decodificados. El dispositivo de visualización 32 puede comprender varios dispositivos de visualización, tales como una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos emisores de luz (OLED) u otro tipo de dispositivo de visualización.

25 [0053] Esta divulgación puede referirse en general al codificador de vídeo 20 que "señaliza" o "transmite" cierta información a otro dispositivo, tal como el decodificador de vídeo 30. El término "señalizar" o "transmitir" puede referirse en general a la comunicación de elementos sintácticos y/u otros datos usados para decodificar los datos de vídeo comprimidos. Dicha comunicación puede producirse en tiempo real o casi real. De forma alternativa, dicha comunicación puede producirse durante un tramo de tiempo, tal como podría producirse cuando se almacenan elementos sintácticos en un medio de almacenamiento legible por ordenador en un flujo de bits codificado en el momento de la codificación, que entonces pueden recuperarse por un dispositivo de descodificación en cualquier momento tras haberse almacenado en este medio. Por lo tanto, aunque el decodificador de vídeo 30 puede denominarse "recibir" cierta información, la recepción de información no necesariamente se produce en tiempo real o casi real y puede recuperarse de un medio en algún momento después del almacenamiento.

30 [0054] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden, cada uno, implementarse como cualquiera entre varios circuitos adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables por campo (FPGA), lógica discreta, hardware o cualquier combinación de los mismos. Si las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado de almacenamiento no transitorio legible por ordenador, y puede ejecutar las instrucciones en hardware usando uno o más procesadores para realizar las técnicas de esta divulgación. Cualquiera de los anteriores (incluyendo hardware, software, una combinación de hardware y software, etc.) puede considerarse como uno o más procesadores. Cada uno del codificador de vídeo 20 y del 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 respectivo dispositivo.

35 [0055] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 funcionan de acuerdo con una norma de compresión de vídeo, tal como la norma HEVC mencionada anteriormente y descrita en el Borrador 10 de la HEVC. Además de la norma base de HEVC, hay esfuerzos constantes para producir extensiones de codificación de vídeo ajustable a escala, codificación de vídeo multivistas y codificación 3D para la HEVC. Además, pueden proporcionarse modalidades de codificación basada en paletas, por ejemplo, como se describe en esta divulgación, para la extensión de la norma HEVC. En algunos ejemplos, las técnicas descritas en esta divulgación para la codificación basada en paletas se pueden aplicar a codificadores y decodificadores configurados para funcionar de acuerdo con otras normas de codificación de vídeo, tales como la norma ITU-TH.264/AVC o normas futuras. Por consiguiente, la aplicación de una modalidad de codificación basada en paletas para la codificación de unidades de codificación (CU) o unidades de predicción (PU) en un códec de HEVC se describe para los propósitos de ejemplo.

40 [0056] En la HEVC y en otras normas de codificación de vídeo, una secuencia de vídeo incluye típicamente una serie de imágenes. Las imágenes también pueden denominarse "tramas". Una imagen puede incluir tres matrices de

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

5 **[0057]** 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 luminancia, dos bloques de árbol de codificación de muestras de crominancia correspondientes y estructuras sintácticas usadas para codificar las muestras de los bloques de árbol de codificación. Un bloque de árbol de codificación puede ser un bloque de muestras de tamaño $N \times N$. Una CTU también puede denominarse "bloque de árbol" o "máxima unidad de codificación" (LCU). Las CTU de la HEVC pueden ser análogas en gran medida a los macrobloques de otras normas, tales como H.264/AVC. Sin embargo, una CTU no está limitada necesariamente a un tamaño particular y puede incluir una o más unidades de codificación (CU). Un fragmento puede incluir un número entero de CTU ordenadas consecutivamente en el escaneo de cuadrado. Un fragmento codificado puede comprender un encabezado de fragmento y datos de fragmento. El encabezado de fragmento de un fragmento puede ser una estructura sintáctica que incluya elementos sintácticos que proporcionen información sobre el fragmento. Los datos del fragmento pueden incluir CTU codificadas del fragmento.

20 **[0058]** Esta divulgación puede usar el término "unidad de vídeo", "bloque de vídeo" o "bloque" para referirse a uno o más bloques de muestras y estructuras sintácticas usadas para codificar muestras del uno o más bloques de muestras. Los tipos de ejemplo de unidades de vídeo pueden incluir CTU, CU, PU, unidades de transformadas (TU), macrobloques, particiones de macrobloque, etc. En algunos contextos, el análisis de PU puede intercambiarse con el análisis de macrobloques o de particiones de macrobloques.

25 **[0059]** Para generar una CTU codificada, el codificador de vídeo 20 puede realizar de forma recursiva una partición de árbol cuaternario en los bloques de árbol de codificación de una CTU para dividir los bloques de árbol de codificación en bloques de codificación, de ahí el nombre "unidades de árbol de codificación". Un bloque de codificación es un bloque $N \times N$ de muestras. Una CU puede ser un bloque de codificación de muestras de luminancia y dos bloques de codificación correspondientes de muestras de crominancia de una imagen que tenga una matriz de muestras de luminancia, una matriz de muestras de Cb y una matriz de muestras de Cr y estructuras sintácticas usadas para codificar las muestras de los bloques de codificación. El codificador de vídeo 20 puede dividir un bloque de codificación de una CU en uno o más bloques predictivos. Un bloque predictivo puede ser un bloque rectangular (es decir, cuadrado o no cuadrado) de muestras en las que se aplique la misma predicción. Una unidad de predicción (PU) de una CU puede ser un bloque predictivo de muestras de luma, dos bloques predictivos correspondientes de muestras de croma de una imagen y estructuras sintácticas usadas para predecir las muestras de bloques predictivos. El codificador de vídeo 20 puede generar bloques predictivos de luma, Cb y Cr para bloques predictivos de luma, Cb y Cr de cada PU de la CU.

40 **[0060]** El codificador de vídeo 20 puede usar intrapredicción o interpredicción para generar los bloques predictivos para una PU. Si el codificador de vídeo 20 usa intrapredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras decodificadas de la imagen asociada a la PU.

45 **[0061]** Si el codificador de vídeo 20 usa interpredicción para generar los bloques predictivos de una PU, el codificador de vídeo 20 puede generar los bloques predictivos de la PU basándose en muestras decodificadas de una o más imágenes distintas a la imagen asociada a la PU. El codificador de vídeo 20 puede usar unipredicción o bipredicción para generar los bloques predictivos de una PU. Cuando el codificador de vídeo 20 usa la unipredicción para generar los bloques predictivos para una PU, la PU puede tener un único vector de movimiento (MV). Cuando el codificador de vídeo 20 usa bipredicción para generar los bloques predictivos para una PU, la PU puede tener dos MV.

50 **[0062]** Después de que el codificador de vídeo 20 genera bloques predictivos (por ejemplo, bloques predictivos de luma, Cb y Cr) para una o más PU de una CU, el codificador de vídeo 20 puede generar bloques residuales para la CU. Cada muestra en un bloque residual de la CU puede indicar una diferencia entre una muestra en un bloque predictivo de una PU de la CU y una muestra correspondiente en un bloque de codificación de la CU. Por ejemplo, el codificador de vídeo 20 puede generar un bloque residual de luma para la CU. Cada muestra en el bloque residual de luma de la CU indica una diferencia entre una muestra de luma en uno de los bloques predictivos de luma de la CU y una muestra correspondiente en el bloque de codificación de luma original de la CU. Además, el codificador de vídeo 20 puede generar un bloque residual de Cb para la CU. Cada muestra en el bloque residual de Cb de la CU puede indicar una diferencia entre una muestra de Cb en uno de los bloques predictivos de Cb de la CU y una muestra correspondiente en el bloque de codificación de Cb original de la CU. El codificador de vídeo 20 también puede generar un bloque residual de Cr para la CU. Cada muestra en el bloque residual de Cr de la CU puede indicar una diferencia entre una muestra de Cr en uno de los bloques predictivos de Cr de la CU y una muestra correspondiente en el bloque de codificación de Cr original de la CU.

65 **[0063]** Además, el codificador de vídeo 20 puede usar la partición en árbol cuádruple para descomponer los bloques residuales (por ejemplo, los bloques residuales de luma, Cb y Cr) de una CU en uno o más bloques de transformada

(por ejemplo, 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 usadas para transformar las muestras de bloques de transformada. De este modo, cada TU de una CU puede estar asociada a un bloque de transformada de luma, un bloque de transformada de Cb y un bloque de transformada de Cr. El bloque de transformada de luma asociado a la TU puede ser un subbloque del bloque residual de luma de la CU. El bloque de transformada de Cb puede ser un subbloque del bloque residual de Cb de las CU. El bloque de transformada de Cr puede ser un subbloque del bloque residual de Cr de la CU.

[0064] El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada para generar un bloque de coeficientes para una TU. Un bloque de coeficientes puede ser una matriz bidimensional de coeficientes de transformada. Un coeficiente de transformada puede ser una cantidad escalar. Por ejemplo, el codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de luma de una TU para generar un bloque de coeficientes de luma para la TU. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cb de una TU para generar un bloque de coeficientes de Cb para la TU. El codificador de vídeo 20 puede aplicar una o más transformadas a un bloque de transformada de Cr de una TU para generar un bloque de coeficientes de Cr para la TU.

[0065] Después de generar un bloque de coeficientes (por ejemplo, un bloque de coeficientes de luma, un bloque de coeficientes de Cb o un bloque de coeficientes de Cr), el codificador de vídeo 20 puede cuantizar el bloque de coeficientes. La cuantización se refiere, en general, a un proceso en el que los coeficientes de transformada se cuantizan 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 cuantice un bloque de coeficientes, el codificador de vídeo 20 puede codificar por entropía elementos sintácticos que indiquen los coeficientes de transformada cuantizados. Por ejemplo, el codificador de vídeo 20 puede realizar la Codificación Aritmética Binaria Adaptativa al Contexto (CABAC) en los elementos sintácticos que indiquen los coeficientes de transformada cuantizados. El codificador de vídeo 20 puede emitir los elementos sintácticos codificados por entropía, en un flujo de bits. El flujo de bits también puede incluir elementos sintácticos que no estén codificados por entropía.

[0066] El codificador de vídeo 20 puede emitir un flujo de bits que incluya los elementos sintácticos codificados por entropía. El flujo de bits puede incluir una secuencia de bits que forme una representación de imágenes codificadas y datos asociados. El flujo de bits puede comprender una secuencia de unidades de capa de abstracción de red (NAL). Cada una de las unidades NAL incluye un encabezado de unidad NAL y encapsula una carga útil de secuencia de octetos sin procesar (RBSP). El encabezado de unidad NAL puede incluir un elemento sintáctico que indique un código de tipo de unidad NAL. El código de tipo de unidad NAL especificado por el encabezado de unidad NAL de una unidad NAL indica el tipo de la unidad NAL. Una RBSP puede ser una estructura sintáctica que contenga un número entero de bytes que se encapsule dentro de una unidad NAL. En algunos casos, una RBSP incluye cero bits.

[0067] Diferentes tipos de unidades NAL pueden encapsular diferentes tipos de las 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 información de realce complementaria (SEI), y así sucesivamente. 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 la capa de codificación de vídeo (VCL).

[0068] El decodificador de vídeo 30 puede recibir un flujo de bits generado por el codificador de vídeo 20. Además, el decodificador de vídeo 30 puede obtener elementos sintácticos del flujo de bits. Por ejemplo, el decodificador de vídeo 30 puede analizar el flujo de bits para decodificar elementos sintácticos del flujo de bits. El decodificador de vídeo 30 puede reconstruir las imágenes de los datos de vídeo basándose al menos en parte en los elementos sintácticos obtenidos (por ejemplo, decodificados) a partir del flujo de bits. El proceso para reconstruir los datos de vídeo puede ser en general recíproco al proceso realizado por el codificador de vídeo 20. Por ejemplo, el decodificador de vídeo 30 puede usar MV de PU para determinar bloques predictivos de muestras (es decir, bloques predictivos) para las PU de una CU actual. Además, el decodificador de vídeo 30 puede cuantizar inversamente bloques de coeficientes de transformada, asociados a las TU de la CU actual. El decodificador de vídeo 30 puede realizar transformadas inversas en los bloques de coeficientes de transformada para reconstruir los bloques de transformada asociados a las TU de la CU actual. El decodificador de vídeo 30 puede reconstruir los bloques de codificación de la CU actual añadiendo las muestras de los bloques predictivos para las PU de la CU actual a las 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 decodificador de vídeo 30 puede reconstruir la imagen.

[0069] En algunos ejemplos, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden configurarse para realizar una codificación basada en paletas. Por ejemplo, en la codificación basada en paletas, en lugar de realizar las técnicas de codificación intrapredictiva o interpredictiva descritas anteriormente, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden codificar una denominada paleta como una tabla de colores o de valores de píxeles que representen los datos de vídeo de un área particular (por ejemplo, un bloque dado). De esta manera, en lugar de codificar los valores de píxeles reales o sus residuales para un bloque actual de datos de vídeo, el codificador de vídeo

puede codificar valores de índice para uno o más de los valores de píxeles del bloque actual, donde los valores de índice indican entradas en la paleta que se usan para representar los valores de píxeles del bloque actual.

[0070] Por ejemplo, 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 que tenga un valor representativo del valor de uno o más píxeles individuales del bloque, y codificando el bloque con un índice valores que indiquen la entrada en la paleta usada para representar uno o más valores de píxeles individuales del bloque. Además, el codificador de vídeo 20 puede señalar los valores de índice en un flujo de bits codificado. A su vez, un dispositivo de decodificación de vídeo (por ejemplo, un decodificador de vídeo 30) puede obtener, a partir del flujo de bits codificado, la paleta para un bloque, así como los valores de índice usados para determinar los distintos píxeles individuales del bloque que usa la paleta. El decodificador de vídeo 30 puede relacionar los valores de índice de los píxeles individuales con las entradas de paleta para reconstruir los valores de píxeles del bloque. En los casos en que el valor de índice asociado con un píxel individual no coincide con ningún valor de índice de la paleta correspondiente para el bloque, el decodificador de vídeo 30 puede identificar dicho píxel como un píxel de escape, para los propósitos de la codificación basada en paletas.

[0071] En otro ejemplo, el codificador de vídeo 20 puede codificar un bloque de datos de vídeo de acuerdo con las siguientes operaciones. El codificador de vídeo 20 puede determinar los valores residuales de predicción para píxeles individuales del bloque, determinar una paleta para el bloque y ubicar una entrada (por ejemplo, un valor de índice) en la paleta que tenga un valor representativo del valor de uno o más de los valores de predicción residual de los píxeles individuales. Adicionalmente, el codificador de vídeo 20 puede codificar el bloque con valores de índice que indiquen la entrada en la paleta usada para representar el valor residual de predicción correspondiente para cada píxel individual del bloque. El decodificador de vídeo 30 puede obtener, a partir de un flujo de bits codificado, señalado por el dispositivo fuente 12, una paleta para un bloque, así como valores de índice para los valores residuales de predicción correspondientes a los píxeles individuales del bloque. Como se describe, los valores de índice pueden corresponder a entradas en la paleta asociada con el bloque actual. A su vez, el decodificador de vídeo 30 puede relacionar los valores de índice de los valores residuales de predicción con las entradas de la paleta para reconstruir los valores residuales de predicción del bloque. Los valores residuales de predicción pueden añadirse a los valores de predicción (por ejemplo, obtenidos usando intra o interpredicción) para reconstruir los valores de píxeles del bloque.

[0072] Como se describe con más detalle a continuación, la idea básica de la codificación basada en paletas es que, para un bloque dado de datos de vídeo que se vaya a codificar, el codificador de vídeo 20 puede derivar una paleta que incluya los valores de píxeles más dominantes en el bloque actual. Por ejemplo, la paleta puede referirse a varios valores de píxeles que se supone que son dominantes y/o representativos de la CU actual. El codificador de vídeo 20 puede transmitir primero el tamaño y los elementos de la paleta al decodificador de vídeo 30. Adicionalmente, el codificador de vídeo 20 puede codificar los valores de píxeles en el bloque dado de acuerdo con un cierto orden de escaneo. Para cada píxel incluido en el bloque dado, el codificador de vídeo 20 puede señalar el valor de índice que asigna el valor de píxel a una entrada correspondiente en la paleta. Si el valor de píxel no está incluido en la paleta (es decir, no existe una entrada de paleta que especifique un valor de píxel particular del bloque codificado por paleta), dicho píxel se define como un "píxel de escape". De acuerdo con la codificación basada en paletas, el codificador de vídeo 20 puede codificar y señalar un valor de índice que esté reservado para un píxel de escape. En algunos ejemplos, el codificador de vídeo 20 también puede codificar y señalar el valor de píxel o un valor residual (o versiones cuantizadas del mismo) para un píxel de escape incluido en el bloque dado.

[0073] Al recibir el flujo de bits de vídeo codificado señalado por el codificador de vídeo 20, el decodificador de vídeo 30 puede determinar primero la paleta basándose en la información recibida del codificador de vídeo 20. El decodificador de vídeo 30 puede asignar entonces los valores de índice recibidos asociados con las ubicaciones de píxeles en el bloque dado a las entradas de la paleta para reconstruir los valores de píxeles del bloque dado. En algunos casos, el decodificador de vídeo 30 puede determinar que un píxel de un bloque codificado por paleta es un píxel de escape, tal como determinando que el píxel está codificado por paleta con un valor de índice reservado para los píxeles de escape. En los casos en que el decodificador de vídeo 30 identifica un píxel de escape en un bloque codificado por paleta, el decodificador de vídeo 30 puede recibir el valor de píxel o un valor residual (o versiones cuantizadas del mismo) para un píxel de escape incluido en el bloque dado. El decodificador de vídeo 30 puede reconstruir el bloque codificado por paleta asignando los valores de píxeles individuales a las entradas correspondientes de la paleta y usando el valor de píxel o el valor residual (o sus versiones cuantizadas) para reconstruir cualquier píxel de escape incluido en el bloque codificado por paleta.

[0074] La codificación basada en paletas puede introducir una cantidad de sobrecarga de señalización. Por ejemplo, se pueden necesitar varios bits para señalar las características de una paleta, tales como el tamaño de la paleta, así como la paleta misma. Además, se pueden necesitar varios bits para indicar valores de índice para los píxeles del bloque. Por ejemplo, de acuerdo con las técnicas de codificación basada en paletas existentes, incluso en los casos en que la paleta tenga un tamaño limitado (por ejemplo, una paleta que incluya solo una entrada), y un bloque no incluya ningún píxel de escape, el codificador de vídeo 20 aún puede indicar índices de paleta para los valores de píxeles del bloque, que serán todos el mismo valor de índice que identifique la entrada en la paleta, línea por línea para el bloque de vídeo. Adicionalmente, las técnicas de codificación basada en paletas existentes introducen la sobrecarga de señalización en términos de señalar un valor de índice para indicar un píxel de escape, y entonces señalar el valor de píxel o el valor residual (o sus versiones cuantizadas) para el píxel de escape.

[0075] Las técnicas de esta divulgación pueden, en algunos ejemplos, reducir el número de bits necesarios para señalar dicha información. Por ejemplo, ciertas técnicas descritas en el presente documento están dirigidas a omitir la codificación (por ejemplo, codificación y/o decodificación) de un mapa de valores de índice para uno o más píxeles de un bloque si el bloque cumple con un conjunto particular de condiciones con respecto a la paleta de codificación. En otros ejemplos, ciertas técnicas descritas en el presente documento están dirigidas en general a señalar menos bits de datos (por ejemplo, un indicador de un bit en lugar de un valor de índice de cinco bits) para indicar que un píxel dado es un píxel de escape con respecto a la paleta para el bloque actual. Diversas técnicas de esta divulgación también están dirigidas a determinar un rango de valores de píxeles que pueden incluirse en una paleta para un bloque dado. El rango de valores de píxeles que puede incluir una paleta se denomina en el presente documento "límite de error" de la paleta, y diversas técnicas de esta divulgación se dirigen a determinar el límite de error de una paleta basándose en el parámetro de cuantización (QP) del bloque asociado con la paleta.

[0076] Otros aspectos de esta divulgación están dirigidos a derivar valores de escape cuantizados. Por ejemplo, algunos de estos aspectos se dirigen a técnicas para definir uno o más parámetros de cuantización con los que cuantizar un píxel de escape. Otros aspectos más de esta divulgación se dirigen a la aplicación de funciones particulares (por ejemplo, una función de desplazamiento hacia la derecha) para cuantizar un valor de píxel de escape. De esta manera, diversos aspectos de esta divulgación proporcionan ventajas potenciales, tal como reducir la sobrecarga de bits y mitigar el uso de recursos, al tiempo que se mantiene la calidad y la precisión de la imagen.

[0077] Como se describió anteriormente, el codificador de vídeo 20 puede aplicar técnicas de esta divulgación para derivar la codificación y la señalización de un índice de paleta para diversos píxeles individuales de un bloque en ciertas circunstancias. De acuerdo con los aspectos de esta divulgación, el codificador de vídeo 20 puede evitar la codificación y la señalización del índice de paleta para un bloque codificado por paleta si el codificador de vídeo 20 determina que todos los píxeles del bloque son del mismo color. Por ejemplo, el codificador de vídeo 20 puede determinar que una CU codificada por paleta de una imagen es una "CU de un único color" y puede derivar la codificación y la señalización del índice de la paleta para la CU de un único color.

[0078] Más específicamente, el codificador de vídeo 20 puede determinar si una CU codificada por paleta es una CU de un único color si la CU actual cumple con dos condiciones. La primera condición que el codificador de vídeo 20 puede usar para determinar si la CU codificada por paleta es una CU de un único color es si el tamaño de la paleta correspondiente es igual a uno (1). Si el tamaño de la paleta es igual a uno, entonces el codificador de vídeo 20 puede determinar que la primera condición se cumple con respecto a que la CU codificada por paleta es una CU de un único color. Más específicamente, si el tamaño de la paleta es uno, entonces el decodificador de vídeo 20 puede determinar que la paleta incluye solo un color que corresponde a los píxeles (no de escape) de la CU codificada por paleta. En algunos ejemplos donde el tamaño de la paleta es uno, el codificador de vídeo 20 puede determinar que el único valor de índice asociado con la paleta es cero (0).

[0079] Si el codificador de vídeo 20 determina que el tamaño de paleta para el CU codificado por paleta es 1 (es decir, que se cumple con la primera condición), el codificador de vídeo 20 puede determinar si el CU codificado por paleta cumple con una segunda condición para ser una CU de un único color. La segunda condición que el decodificador de vídeo 30 puede usar para determinar si la CU codificada por paleta es una CU de un único color es que la CU codificada por paleta no incluye ningún píxel de escape. Si la CU codificada por paleta incluye al menos un píxel de escape, el codificador de vídeo 20 puede determinar entonces que, aunque la paleta correspondiente solo indica un color con respecto a la CU codificada por paleta, la CU codificada por paleta incluye píxeles de dos o más colores. Por ejemplo, la CU codificada por paleta puede incluir al menos un píxel que tenga el color indicado en la paleta, y al menos un píxel de escape que tenga un color diferente.

[0080] Si el codificador de vídeo 20 determina que la CU codificada por paleta cumple con las dos condiciones descritas anteriormente, entonces el codificador de vídeo 20 puede determinar que la CU codificada por paleta es una CU de un único color. Más específicamente, si la CU codificada por paleta se asocia con una paleta de entrada única (que se muestra por el tamaño de paleta de uno), y la CU codificada por paleta no incluye ningún píxel de escape, entonces el codificador de vídeo 20 puede determinar que todos los píxeles individuales de las CU codificadas por paleta son del mismo color (es decir, el color indicado por la entrada única de la paleta correspondiente). En diversas implementaciones, el codificador de vídeo 20 puede aplicar las técnicas de identificación de CU de un único color descritas anteriormente con respecto a una base de componente de un único color con respecto a la paleta, o a un índice combinado que indique más de un componente de color.

[0081] En respuesta a determinar que la CU codificada por paleta incluye píxeles de solo un color (es decir, que la CU es una CU de un único color), el codificador de vídeo 20 puede implementar técnicas de esta divulgación para derivar la codificación y la señalización del mapa de valores de índice de paleta para los píxeles de una CU de un único color. Al omitir la codificación y la señalización del mapa de valores de índice de paleta para los píxeles de la CU, el codificador de vídeo 20 puede conservar los recursos informáticos y el ancho de banda que de otra manera se utilizaría para codificar y señalar la información de color para la CU. En lugar de codificar y señalar un índice para cada píxel individual de la CU de un único color, el codificador de vídeo 20 puede implementar las técnicas de esta divulgación para indicar más eficientemente (por ejemplo, al decodificador de vídeo 30) la información de color para

todo el conjunto de píxeles que forman la CU de un único color.

[0082] De acuerdo con algunos ejemplos de esta divulgación, el codificador de vídeo 20 puede codificar y señalar un indicador para indicar si el codificador de vídeo 20 omitió la codificación (y la señalización) de los valores de índice de paleta línea por línea para la CU de un único color. Al codificar y señalar un indicador de un bit para la CU en lugar de valores de índice individuales para cada píxel de la CU, el codificador de vídeo 20 puede conservar los recursos informáticos y el ancho de banda de señalización en comparación con las técnicas de codificación basada en paletas existentes. Además, el codificador de vídeo 20 puede mantener la precisión y la calidad de la CU codificada de un único color, porque la paleta de una sola entrada señalada por el codificador de vídeo 20 para la CU incluye la información de color para todos los píxeles individuales de la CU. En diversos ejemplos, el codificador de vídeo 20 puede codificar y señalar el indicador de diversas formas, tal como en un conjunto de parámetros de secuencia (SPS), un conjunto de parámetros de imagen (PPS) o un encabezado de fragmento. En diversos ejemplos, el codificador de vídeo 20 puede codificar y señalar el indicador por CTU, por CU o también por un bloque de cualquier tamaño de bloque.

[0083] En los ejemplos en los que el codificador de vídeo 20 omite la codificación y la señalización del valor de índice de paleta para píxeles individuales de un bloque de un único color que esté codificado por paleta, el decodificador de vídeo 30 puede aplicar diversas técnicas de esta divulgación para reconstruir el bloque de un único color. En algunos ejemplos, el decodificador de vídeo 30 puede realizar operaciones recíprocas a las descritas anteriormente con respecto al codificador de vídeo 20 para determinar que el bloque codificado por paleta es un bloque de un único color. Por ejemplo, el decodificador de vídeo 30 puede determinar que la paleta para el bloque actual tiene un tamaño de uno, determinando de este modo que el bloque cumple con la primera condición para clasificarse como un bloque de un único color. En diversos ejemplos, el decodificador de vídeo 30 puede recibir la paleta en un flujo de bits de vídeo codificado del codificador de vídeo 20, o puede reconstruir la paleta.

[0084] Adicionalmente, el decodificador de vídeo 30 puede determinar que el bloque no incluye ningún píxel de escape, lo que determina que el bloque cumpla con la segunda condición para clasificarse como un bloque de un único color. Basándose en determinar que el tamaño de la paleta para el bloque es uno (la primera condición) y que el bloque no incluye ningún píxel de escape (la segunda condición), el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para determinar que el bloque actual es un bloque de un único color. De esta manera, el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para reconstruir un bloque codificado por paleta con precisión, al tiempo que conserva los recursos informáticos y el ancho de banda a los que, por el contrario, se les requeriría reconstruir el bloque al decodificar un índice de paleta en una base de píxel a píxel.

[0085] En otros ejemplos, el decodificador de vídeo 30 puede recibir, en el flujo de bits de vídeo codificado, un indicador que indica si el codificador de vídeo 20 omitió la codificación y la señalización del índice de paleta para un bloque codificado por paleta, de acuerdo con las técnicas de esta divulgación. En los casos en que el decodificador de vídeo 30 reciba un indicador que indique que el codificador de vídeo 20 omitió la codificación y la señalización del mapa de valores de índice de paleta para el bloque codificado por paleta, el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para determinar que el bloque actual está codificado por paleta, y es un bloque de un único color. Más específicamente, si el indicador está habilitado (por ejemplo, establecida en un valor de uno), el decodificador de vídeo 30 puede determinar que el bloque codificado por paleta es un bloque de un único color. A su vez, el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para reconstruir cada píxel del bloque de acuerdo con la información de color de la única entrada en la paleta para el bloque. De esta manera, el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para reconstruir con precisión el bloque codificado por paleta usando un indicador de un bit para todo el bloque, en lugar de usar valores de índice separados (con una profundidad de bits variable) para cada píxel individual del bloque.

[0086] En otro ejemplo, el codificador de vídeo 20 puede implementar técnicas de esta divulgación para derivar un límite de error para una paleta para un bloque codificado por paleta. Como se usa en el presente documento, los términos "límite de error" o "límite de error de paleta" pueden referirse al rango de valores (por ejemplo, en términos de información de color) que pueden incluir las entradas de la paleta. Más específicamente, el límite de error de paleta define una variación mínima en el valor del color que las diferentes entradas de la paleta tratan o deben mostrar. Como se describió anteriormente, con el fin de codificar un bloque de acuerdo con la codificación basada en paleta, el codificador de vídeo 20 puede construir la paleta correspondiente para incluir valores de color que se produzcan con mayor frecuencia (píxel por píxel) dentro del bloque.

[0087] Al construir una paleta, el codificador de vídeo 20 puede determinar que las distintas entradas de la paleta deben mostrar una variación mínima entre sí. Más específicamente, el codificador de vídeo 20 puede construir la paleta para que no haya dos entradas de la paleta que sean lo suficientemente similares de modo que las dos entradas se puedan agrupar como una sola entrada. Si dos entradas posibles de la paleta están dentro del límite de error de la paleta, el codificador de vídeo 20 puede usar una de las dos entradas para representar ambas entradas en la paleta.

[0088] Sin embargo, si el codificador de vídeo 20 determina que dos entradas (que se producen comúnmente en el bloque) difieren al menos en el límite de error de la paleta, entonces el codificador de vídeo 20 puede incluir ambas entradas en la paleta. En ejemplos donde las entradas están representadas por tres componentes de color, el

codificador de vídeo 20 puede incluir ambas entradas en la paleta si en las entradas difieren al menos en el límite de error de paleta con respecto a al menos uno de los componentes de color. Por ejemplo, si el límite de error de paleta se establece en un valor de cinco (5), el codificador de vídeo 20 puede determinar si incluir ambas entradas en la paleta (suponiendo que ambas entradas se produzcan con suficiente frecuencia en el bloque), basándose en la siguiente expresión booleana : $\text{abs}(A1-B1) > 5 \parallel \text{abs}(A2-B2) > 5 \parallel \text{abs}(A3-B3) > 5$, donde "abs" representa una diferencia entre los parámetros de los componentes de color.

[0089] Como se describe, el codificador de vídeo 20 puede construir una paleta agrupando los valores de píxeles que se producen comúnmente (o que se producen relativamente frecuentemente) del bloque en entradas de la paleta. El codificador de vídeo 20 puede seleccionar los valores de píxeles que se producen comúnmente de modo que los valores de píxeles muestren una variación mínima, en términos de información de color. A su vez, la variación mínima entre los valores de píxeles dentro del conjunto seleccionado de valores de píxeles comunes puede formar el límite de error de la paleta correspondiente. Se apreciará que, aunque el límite de error de la paleta puede incluir varios valores de píxeles, es posible que la paleta no incluya necesariamente todos los pares de valores de píxeles que difieran al menos en el límite de error de paleta. Por tanto, el mismo límite de error de paleta puede aplicarse a paletas de diferentes tamaños. El codificador de vídeo 20 puede usar el límite de error de paleta para determinar los valores de color que se incluirán en la paleta.

[0090] El codificador de vídeo 20 puede implementar técnicas de esta divulgación para definir el límite de error para una paleta. De acuerdo con diversos aspectos de esta divulgación, el codificador de vídeo 20 puede determinar el límite de error de paleta basado en el parámetro de cuantización (QP) para el bloque codificado por paleta. En diversos ejemplos, el codificador de vídeo 20 puede determinar que el límite de error de paleta es directamente proporcional al valor QP para el bloque correspondiente. Más específicamente, en estos ejemplos, el codificador de vídeo 20 puede asignar un límite de error mayor para una paleta para un bloque que se cuantiza con un valor de QP mayor, y un límite de error menor para una paleta para un bloque que se cuantiza con un valor de QP menor.

[0091] Por tanto, el codificador de vídeo 20 puede definir paletas que requieran una mayor variación entre los valores de píxeles para los bloques que se cuantizan con valores de QP mayores, y puede definir paletas que requieran una variación menor entre los valores de píxeles para los bloques que se cuantizan con valores de QP menores. Además, el codificador de vídeo 20 puede generar y/o almacenar una tabla (por ejemplo, una tabla de asignación o una tabla de consulta) para reflejar la relación entre cada valor de QP y el límite de error de paleta correspondiente. De esta manera, el codificador de vídeo 20 puede implementar varias técnicas de esta divulgación para mejorar la eficiencia computacional mediante el uso de una tabla para almacenar la relación entre cada valor de QP y el límite de error correspondiente. Más específicamente, al usar una tabla para almacenar la relación entre los valores de QP y el límite de error de paleta correspondiente, el codificador de vídeo 20 puede implementar las técnicas descritas en el presente documento para proporcionar una eficiencia mejorada en comparación con las técnicas computacionalmente costosas de resolver una función para cada paleta para derivar el límite de error de paleta correspondiente. Por tanto, el codificador de vídeo 20 puede personalizar la paleta (de acuerdo con el límite de error de paleta) basándose en el valor QP con el que se cuantiza el bloque correspondiente, determinando de este modo los contenidos de la paleta para un bloque basado en el valor QP del bloque, de acuerdo con diversos aspectos de esta divulgación.

[0092] El codificador de vídeo 20 puede, en algunos ejemplos, implementar diversas técnicas de esta divulgación para la derivación cuantizada de píxeles de escape. Más específicamente, el codificador de vídeo 20 puede implementar las técnicas para definir el valor de cuantización del QP para un píxel de escape. Por ejemplo, de acuerdo con las técnicas de codificación basada en paletas, si el codificador de vídeo 20 detecta un píxel de escape en un bloque codificado por paleta, el codificador de vídeo 20 puede codificar y señalar el valor de píxel, o un error de predicción del mismo, porque la paleta correspondiente no incluye cualquier entrada para el píxel de escape. Adicionalmente, para conservar el ancho de banda de señalización, el codificador de vídeo 20 puede cuantizar el valor de píxel codificado del píxel de escape antes de la señalización.

[0093] De acuerdo con las técnicas de codificación basada en paletas existentes, no se definió ningún valor de cuantización (valor de QP) para cuantizar un píxel de escape. El codificador de vídeo 20 puede implementar las técnicas de esta divulgación para definir el valor QP para cuantizar un píxel de escape. Más específicamente, el codificador de vídeo 20 puede definir el valor de QP para un píxel de escape igual al valor de QP para la codificación de coeficiente tradicional dentro del mismo canal de color (por ejemplo, luma (Y), croma (U, Cr) o croma (V, Cb)). En un ejemplo, el codificador de vídeo 20 puede definir el valor de QP para un píxel de escape igual al valor de QP para la codificación de coeficiente tradicional dentro del mismo canal de color y dentro del mismo grupo de cuantización. Por tanto, el codificador de vídeo 20 puede cuantizar todos los píxeles de escape de acuerdo con un único valor de QP dentro de un canal dado. Adicionalmente, como el codificador de vídeo 20 puede definir el valor de QP para todos los píxeles de escape solo dentro de un único canal, el codificador de vídeo 20 puede usar diferentes valores de QP para cuantizar los píxeles de escape con respecto a diferentes canales.

[0094] El decodificador de vídeo 30 puede realizar operaciones recíprocas a las descritas anteriormente, para descuantizar píxeles de escape de acuerdo con diversas técnicas de esta divulgación. Por ejemplo, el decodificador de vídeo 30 puede descuantizar todos los píxeles de escape de un único canal usando el mismo valor QP, basándose en la información recibida en el flujo de bits de vídeo codificado del codificador de vídeo 20. Más específicamente, de

acuerdo con los aspectos de esta divulgación, el decodificador de vídeo 30 puede descuantizar cualquier píxel de escape (o errores de predicción/valores residuales del mismo) comunicados por un canal particular usando un valor de QP que se determina basándose en el valor de QP para la descuantización del coeficiente de transformada tradicional para bloques comunicados por el canal actual. En algunos ejemplos, el decodificador de vídeo 30 puede implementar las técnicas de esta divulgación para descuantizar los píxeles de escape comunicados a través de diferentes canales usando diferentes valores de QP, basándose en el valor de QP para la codificación del coeficiente de transformada tradicional que es diferente entre los diferentes canales.

[0095] De esta manera, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementar las técnicas descritas en el presente documento para definir y aplicar un único valor de QP (para cuantizar y/o descuantizar) todos los píxeles de escape comunicados a través de un canal en particular. Por tanto, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden aplicar aspectos de esta divulgación para definir un valor de QP para los píxeles de escape detectados a través de la codificación basada en paletas, donde las técnicas de codificación existentes basada en paletas no definieron explícitamente un valor de QP para los píxeles de escape.

[0096] Adicionalmente, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden implementar otras técnicas de esta divulgación para usar un indicador para indicar y/o detectar la inclusión de un píxel de escape en un bloque codificado por paleta. De acuerdo con las técnicas de codificación basada en paletas existentes, los píxeles de escape pueden señalizarse y detectarse usando un valor de índice de paleta "reservado". Por ejemplo, de acuerdo con las técnicas de codificación basada en paletas existentes, el valor de índice de paleta reservado que indica un píxel de escape puede ser 32. Más específicamente, el valor de índice de paleta de 32 se puede usar para todos los píxeles de escape, independientemente de si dos píxeles de escape tienen valores de píxeles diferentes. Por lo tanto, de acuerdo con las técnicas de codificación basada en paletas existentes, los dispositivos de codificación de vídeo pueden usar un valor de cinco bits (de 32) para cada píxel de escape de un bloque codificado por paleta.

[0097] El codificador de vídeo 20 puede implementar técnicas de esta divulgación para conservar los recursos informáticos (por ejemplo, almacenamiento y memoria) y reducir el consumo de ancho de banda, mientras mantiene la precisión de la imagen con respecto a la señalización de un píxel de escape en un bloque codificado por paleta. Por ejemplo, el codificador de vídeo 20 puede codificar y señalar un indicador para indicar si un píxel en un bloque codificado por paleta es un píxel de escape. Como se describe en el presente documento, el indicador, cuando está habilitada, puede indicar que al píxel asociado se le asigna un índice de paleta denominado "otro índice". El codificador de vídeo 20 puede usar el estado de "otro índice" del indicador para reemplazar el valor de índice de paleta de 32 que tradicionalmente se usa para indicar un píxel de escape con respecto a la paleta. Por tanto, el codificador de vídeo 20 puede codificar y señalar un indicador de un bit en lugar de un valor de índice de cinco bits para indicar que un píxel de un bloque codificado por paleta es un píxel de escape. A su vez, cuando un píxel de escape está indicado por el indicador de un bit, el codificador de vídeo 20 puede codificar y señalar el valor de píxel (o sus datos residuales) del píxel de escape en el flujo de bits de vídeo codificado.

[0098] El decodificador de vídeo 30 también puede implementar técnicas de esta divulgación para usar un indicador de un bit para determinar que un píxel de un bloque codificado por paleta es un píxel de escape. En diversos ejemplos, el decodificador de vídeo 30 puede realizar operaciones recíprocas con respecto a las operaciones de codificación y señalización descritas anteriormente con respecto al codificador de vídeo 20, para usar el indicador de un bit para identificar un píxel de escape en un bloque codificado por paleta. Por ejemplo, el decodificador de vídeo 30 puede recibir un indicador de un bit habilitado asociado con un píxel de un bloque codificado por paleta. Basándose en el indicador de un bit que está en el estado habilitado, el decodificador de vídeo 30 puede determinar que la información de color para el píxel asociado no se incluye en la paleta para el bloque actual. En otras palabras, el decodificador de vídeo 30 puede determinar que, si el indicador de un bit recibido está habilitado, el píxel asociado es un píxel de escape. De esta manera, el decodificador de vídeo 30 puede implementar las técnicas de esta divulgación para reconstruir un bloque codificado por paleta usando un indicador de un bit para identificar un píxel de escape en el bloque codificado por paleta. Por tanto, el decodificador de vídeo 30 puede conservar los recursos informáticos (por ejemplo, almacenamiento y/o memoria) y los requisitos de ancho de banda con respecto a la identificación de píxeles de escape en bloques codificados por paleta. Adicionalmente, cuando un píxel de escape está indicado por el indicador de un bit, el decodificador de vídeo 30 puede recibir, en el flujo de bits de vídeo codificado, la información de color (o sus datos residuales) correspondiente a cualquier píxel de escape identificado, y puede reconstruir el bloque codificado por paletas en consecuencia.

[0099] El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden implementar técnicas de esta divulgación para cuantizar y descuantizar los valores de píxeles de los píxeles de escape de acuerdo con la codificación basada en paletas. Por ejemplo, el codificador de vídeo 20 puede conservar los recursos informáticos (por ejemplo, el uso de la memoria, los ciclos del reloj del procesador, etc.) cuantizando los valores de píxeles de los píxeles de escape de acuerdo con los aspectos de esta divulgación. En algunos ejemplos, el codificador de vídeo 20 puede implementar las técnicas descritas en el presente documento para cuantizar los valores de píxeles de escape sustituyendo operaciones de división con operaciones de desplazamiento (por ejemplo, operaciones de desplazamiento a la derecha). Más específicamente, el codificador de vídeo 20 puede determinar la operación específica de desplazamiento hacia la derecha basándose en el valor QP del píxel de escape correspondiente.

[0100] Por ejemplo, el codificador de vídeo 20 puede formar una tabla que asigne el valor QP de cada píxel de escape a la cantidad del desplazamiento a la derecha para aplicar al valor del píxel. El codificador de vídeo 20 puede formar la tabla para incluir 52 entradas. Por ejemplo, la tabla de asignación de 52 entradas puede proporcionar una cantidad de desplazamiento a la derecha correspondiente a cada posible valor de QP para un píxel de escape dado. De forma alternativa, el codificador de vídeo 20 puede aplicar una operación de asignación para determinar la cantidad de desplazamiento a la derecha para cada píxel, basándose en la entrada del valor QP correspondiente en la tabla. La función de asignación puede ser más eficiente computacionalmente y puede conservar los requisitos de memoria, en comparación con la tabla de asignación de 52 entradas usada de acuerdo con las técnicas de cuantización existentes para píxeles de escape de acuerdo con la codificación basada en paletas. Al derivar el valor de desplazamiento a la derecha (operando) resolviendo una función como se describe en el presente documento, el codificador de vídeo 20 puede eliminar la necesidad de que el decodificador de vídeo 30 almacene una tabla de 52 entradas, permitiendo de ese modo al decodificador de vídeo 30 descuantizar píxeles de escape al tiempo que reduce los requisitos de almacenamiento para el proceso de descuantización.

[0101] En diversos ejemplos, el codificador de vídeo 20 puede cuantizar un píxel de escape determinando la cantidad de desplazamiento a la derecha para un píxel de escape basándose en la operación de asignación descrita anteriormente, y aplicando una función lineal al valor del píxel de escape, usando la cantidad de desplazamiento a la derecha determinada. Un ejemplo de una función lineal que el codificador de vídeo 20 puede aplicar para cuantizar un píxel de escape es el siguiente:

$$\text{Right_shift} = a * ((QP + b) \gg c) + d,$$

donde a, b, c y d son todos los parámetros de enteros. Adicionalmente, el operador ">>" denota la operación de desplazamiento a la derecha. En un resultado específico de aplicar la ecuación anterior, el codificador de vídeo 20 puede determinar que la cantidad de desplazamiento a la derecha para un valor de píxel de escape es tres. La operación de desplazamiento a la derecha resultante se puede expresar como $\text{Right_shift} = (QP \gg 3)$

[0102] El decodificador de vídeo 30 puede implementar técnicas de esta divulgación para realizar operaciones recíprocas de las descritas anteriormente con respecto al codificador de vídeo 20, para descuantizar un valor de píxel de escape cuantizado. Por ejemplo, el decodificador de vídeo 30 puede implementar técnicas de esta divulgación para calcular una cantidad de desplazamiento (por ejemplo, para una operación de desplazamiento a la izquierda correspondiente) basada en un valor de QP para descuantizar el valor de píxel de escape cuantizado correspondiente. De esta manera, el decodificador de vídeo 30 también puede aplicar aspectos de esta divulgación para conservar los recursos informáticos al aprovechar una función de asignación en lugar de almacenar una tabla de asignación de 52 entradas.

[0103] Como se describió anteriormente, el codificador de vídeo 20 y/o el decodificador de vídeo 30 pueden implementar diversas técnicas de esta divulgación, ya sea individualmente o en cualquier combinación y/o secuencia, para proporcionar una eficiencia de codificación mejorada con respecto a la codificación basada en paletas, mientras se mantiene la calidad de imagen y precisión de los datos. Por lo tanto, las técnicas descritas en el presente documento pueden proporcionar varias ventajas potenciales sobre las técnicas existentes de codificación de vídeo basada en paletas. En ejemplos específicos, como se describió anteriormente, las técnicas de esta divulgación pueden permitir que los dispositivos de codificación de vídeo codifiquen y/o decodifiquen más eficazmente los datos de vídeo y reduzcan el consumo de ancho de banda, mientras mantienen la precisión de los datos de vídeo.

[0104] En algunos ejemplos, las técnicas para la codificación basada en paletas de datos de vídeo pueden usarse con una o más técnicas de codificación diferentes, tales como técnicas para la codificación inter o intrapredictiva. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o un decodificador, o un codificador-decodificador combinado (códec), se puede configurar para realizar una codificación interpredictiva e intrapredictiva, así como una codificación basada en paletas.

[0105] En diversos ejemplos, esta divulgación describe diferentes aspectos de las principales técnicas de codificación de índice de color. Puede ser posible combinar parte o todos los procedimientos descritos.

[0106] Ahora se describe un ejemplo de codificación de la dirección de predicción de índice. Como se indicó anteriormente, para cada índice, hay tres direcciones de predicción posibles: 'copy_from_top', 'copy_from_left', y 'no_copy'. Se deben asignar tres contraseñas a las tres direcciones. Por ejemplo, las contraseñas pueden ser '0', '01' y '10'. En el caso de que el píxel colocado en la línea anterior y el píxel adyacente izquierdo sean los mismos, solo serán necesarias dos contraseñas. Por ejemplo, en este caso, '0' puede representar 'sin copia' y '1' puede representar copia desde la parte superior o izquierda.

[0107] Como se describió anteriormente, en algunos casos, la codificación del mapa de índice de color para un bloque de vídeo se puede derivar. Si el número de colores principales es igual a uno y no hay un 'píxel de escape', se puede omitir la codificación del bloque de índice. Este principio puede aplicarse a cada componente de color individual o puede aplicarse al índice de combinación que contenga más de un componente de color.

[0108] En otro ejemplo, se puede señalar un indicador (u otro tipo de elemento sintáctico) en el flujo de bits codificado para indicar si esta característica de codificación de índice de omisión se usa o no. Por ejemplo, un codificador de vídeo puede señalar, en un flujo de bits que comprenda una representación codificada de datos de vídeo, un elemento sintáctico (por ejemplo, el indicador) para indicar si se usa o no la codificación de índice de omisión. Por consiguiente, un decodificador de vídeo puede obtener, a partir de un flujo de bits, un elemento sintáctico que indica si se usa o no la codificación de índice de omisión. El indicador se puede señalar en un SPS, PPS, encabezado de fragmento u otra estructura, o por CTU o por CU o en cualquier otro tamaño de bloque.

[0109] Por tanto, en algunos ejemplos, el codificador de vídeo puede señalar, en el flujo de bits, un elemento sintáctico que indica si el bloque de índice está señalado en el flujo de bits. En algunos ejemplos, el codificador de vídeo puede señalar el elemento sintáctico en un SPS, un PPS o un encabezado de fragmento en el flujo de bits. Además, en algunos ejemplos, el codificador de vídeo puede señalar el elemento sintáctico por CTU o por CU. En algunos de estos ejemplos, el decodificador de vídeo puede obtener, a partir del flujo de bits, un elemento sintáctico que indica si el bloque de índice está señalado en el flujo de bits. El indicador se puede señalar en un SPS, PPS, un encabezado de fragmento u otra estructura sintáctica, o por CTU o por CU o en cualquier otro tamaño de bloque. Por tanto, en algunos ejemplos, el decodificador de vídeo puede obtener el elemento sintáctico de un SPS, un PPS o un encabezado de fragmento en el flujo de bits. Además, en algunos ejemplos, el decodificador de vídeo puede obtener el elemento sintáctico por CTU o por CU.

[0110] Ahora se describe un ejemplo de codificación de planos de bits de índices. En el modo de línea normal, si un índice no se puede predecir desde arriba o hacia la izquierda; o en modo horizontal, la línea se copia desde el índice situado más a la izquierda desde la línea actual, el valor del índice debe codificarse directamente. En este caso, un valor de índice se puede codificar bin por bin de acuerdo con una representación binaria del valor del índice. Por ejemplo, suponiendo un índice ubicado en la línea i , la columna j se indica con:

$$C_{i,j} = b_{0ij} + 2b_{1ij} + \dots + 2^N b_{Nij} = [b_{0ij}b_{1ij}\dots b_{Nij}]_2$$

donde $b_{kij} = 0$ o 1 . Entonces, b_{kij} puede codificarse usando valores de índice adyacentes codificados de b_{kij} como los contextos CABAC. Por ejemplo, b_{kij} puede usar $b_{k(i-1)j} + b_{ki(j-1)}$ como contexto. b_{kij} también puede codificarse usando un único contexto, o sin ningún contexto, es decir, evitar la codificación.

[0111] Para permitir un mayor rendimiento, algunas de las bandejas del índice están codificadas en omisión y otras usando contextos CABAC. Por ejemplo, solo la Bandeja más significativa de la representación usa el contexto, mientras que las otras están codificadas en modo omisión.

[0112] Ahora se describe un ejemplo de un indicador para indicar 'otro índice'. En este ejemplo, se puede usar un indicador de un bit para indicar si un índice es 'otro índice' o no. Este indicador se puede codificar usando CABAC con índices adyacentes circundantes codificados del indicador como contexto.

[0113] Ahora se describe un ejemplo de cuantización del valor de escape usando una función para el desplazamiento a la derecha. La tabla para asignar de cada QP a la cantidad de desplazamiento a la derecha requiere 52 entradas. Una función de asignación puede guardar estos requisitos de memoria y proporcionar una manera eficiente de calcular el desplazamiento a la derecha. Por ejemplo, una función lineal podría aplicarse:

$$\text{Right_shift} = a * ((QP + b) \gg c) + d$$

donde a , b , c y d son parámetros enteros. Un ejemplo específico de esta función es el siguiente:

$$\text{Right_shift} = (QP \gg 3)$$

[0114] Ahora se describe un ejemplo de binarización y codificación de índices. En este ejemplo, primero, un indicador se codifica usando los índices codificados adyacentes como contexto para indicar si el índice es cero o no. Si el índice no es cero, supongamos que el índice es $C > 0$. Luego, $C-1$ se binariza y codifica usando la codificación CABAC de omisión. Los ejemplos de procedimientos de binarización incluyen, pero no se limitan a: unario, truncado unario, Golomb exponencial o Golomb-Rice con parámetros fijos o adaptativos.

[0115] A continuación se describe una técnica de ejemplo para evitar el indicador de indicación de píxeles de escape. En un ejemplo, se puede usar un indicador para indicar si un píxel es un 'píxel de escape' (es decir, no se presenta en una tabla de colores principales) o no. Este indicador se puede omitir si el número de colores principales es menor que un número máximo de colores principales, lo que implícitamente indica que no existe un 'píxel de escape'. Este número máximo de colores principales se puede predefinir o ajustar de forma adaptativa. Cuando se deriva el indicador, los datos que indican el indicador no se incluyen en un flujo de bits.

[0116] Por ejemplo, en algunos ejemplos, un codificador de vídeo puede derivar, de un flujo de bits, datos que indiquen las banderas si el número de colores principales para un bloque es menor que un número máximo permitido de colores principales. Por lo tanto, si el número de valores de muestra distintos para los píxeles en el bloque es menor que el número máximo permitido de colores principales, puede haber una entrada en la tabla de colores principales para cada uno de los valores de muestra distintos de los píxeles del bloque y ninguno de los píxeles del bloque es un píxel de escape. A la inversa, si el número de valores de muestra distintos para el píxel en el bloque es mayor que el número máximo permitido de colores principales, uno o más de los píxeles del bloque es un píxel de escape. Por lo tanto, si el número de valores de muestra distintos para el píxel en el bloque es mayor que el número máximo permitido de colores principales, el codificador de vídeo puede señalar indicadores para indicar cuál de los píxeles del bloque son píxeles de escape.

[0117] En un ejemplo, un decodificador de vídeo puede obtener, de un flujo de bits que comprenda una representación codificada de los datos de vídeo, un elemento sintáctico que indique si un píxel en un bloque es un píxel de escape cuando el número de valores de muestra distintos de píxeles en el bloque es mayor que un número máximo permitido de colores en una tabla de colores principales. En este ejemplo, el decodificador de vídeo no obtiene el elemento sintáctico del flujo de bits cuando el número de valores de muestra distintos de píxeles en el bloque es menor que el número máximo permitido de colores en la tabla de colores principales. Cuando el píxel no es un píxel escapado, el decodificador de vídeo puede determinar, basándose en un índice para el píxel, una entrada en la tabla de colores principales que especifique un valor de muestra para el píxel.

[0118] En un ejemplo similar, si el número de valores de muestra distintos de píxeles en un bloque es mayor que el número máximo permitido de colores en una tabla de colores principales, un codificador de vídeo puede incluir, en un flujo de bits que comprenda una representación codificada de los datos de vídeo, datos que indican un elemento sintáctico que indica si un píxel del bloque es un píxel de escape. Si el número de valores de muestra distintos de píxeles en un bloque es menor que el número máximo permitido de colores en una tabla de colores principales, el codificador de vídeo puede omitir el elemento sintáctico del flujo de bits. Cuando el píxel no es un píxel de escape, el codificador de vídeo puede incluir, en el flujo de bits, datos que indiquen un índice que especifique una entrada en la tabla de colores principales que especifica un valor de muestra para el píxel.

[0119] En otro ejemplo, se puede señalar un indicador (u otro tipo de elemento sintáctico) en el flujo de bits codificado para indicar si esta característica de evitar el indicador de indicación de píxeles de escape se usa o no. Por ejemplo, un codificador de vídeo puede señalar, en el flujo de bits codificado, un elemento sintáctico para indicar si se usa o no un elemento sintáctico de indicación (por ejemplo, el indicador de indicación) de píxeles de escape. Por consiguiente, un decodificador de vídeo puede obtener, a partir de un flujo de bits, el elemento sintáctico que indica si se usa o no el elemento sintáctico de indicación de píxeles de escape. El indicador se puede señalar en un SPS, PPS, un encabezado de fragmento u otra estructura, o por CTU o por CU o en cualquier otro tamaño de bloque.

[0120] Por tanto, en algunos ejemplos, un codificador de vídeo puede señalar, en un flujo de bits, un segundo elemento sintáctico que indica si el flujo de bits incluye un primer elemento sintáctico (es decir, un elemento sintáctico que indica si un píxel es un píxel de escape). Además, en algunos ejemplos, el decodificador de vídeo puede obtener, a partir del flujo de bits, un segundo elemento sintáctico que indica si el flujo de bits incluye un primer elemento sintáctico (es decir, un elemento sintáctico que indica si un píxel es un píxel de escape). En algunos ejemplos, este segundo elemento sintáctico puede señalizarse en un conjunto de parámetros de secuencia, un conjunto de parámetros de imagen o un encabezado de fragmento. En algunos ejemplos, el segundo elemento sintáctico se señala por CTU o por CU.

[0121] Ahora se describen ejemplos de procedimientos de codificación por entropía de valores de píxeles de escape cuantizados o errores de predicción de escape cuantizados. En algunos ejemplos, los valores de píxeles de escape cuantizados (errores de predicción) se binarizan usando contraseñas de longitud fija. Para la primera bandeja de la contraseña, la codificación CABAC se aplica con un modelado de contexto. Para las bandejas restantes de la contraseña, la codificación de omisión de CABAC se aplica con probabilidades iguales. En este ejemplo, la longitud de la contraseña depende de los valores de QP para cada canal de luminancia-crominancia (YUV o RGB). Por ejemplo, dada una entrada de datos de profundidad de 8 bits, después de la cuantización del tamaño de paso 4, el valor cuantizado está en el rango de [0, 63] y, por tanto, se puede usar una contraseña de longitud fija de 6 bits en lugar de una contraseña de 8 bits, con el fin de reducir los bits que se vayan a transmitir.

[0122] Por ejemplo, un decodificador de vídeo puede determinar si un píxel de una imagen de los datos de vídeo es un píxel de escape. En respuesta a la determinación de que el píxel no es un píxel escapado, el decodificador de vídeo puede determinar un índice para el píxel y determinar, basándose en el índice para el píxel, una entrada de paleta que especifique un valor de muestra para el píxel. La entrada de paleta puede estar en una paleta que comprenda entradas de paleta que especifiquen valores de muestra. En respuesta a la determinación de que el píxel es un píxel escapado, el decodificador de vídeo puede usar CABAC con modelado de contexto para decodificar la primera casilla de una contraseña de longitud fija. Además, en respuesta a la determinación de que el píxel es un píxel escapado, el decodificador de vídeo puede usar la codificación de omisión CABAC para decodificar por entropía cada bin de la contraseña de longitud fija que sigue al primer bin de la contraseña de longitud fija. Además, en respuesta a la determinación de que el píxel es un píxel escapado, el decodificador de vídeo puede debinarizar la contraseña de

longitud fija para determinar el valor de muestra para el píxel. En algunos ejemplos, la longitud de la contraseña de longitud fija depende de los valores del parámetro de cuantización (QP) para cada canal (por ejemplo, luminancia, crominancia, etc. canal) de la imagen.

5 **[0123]** En un ejemplo similar, un codificador de vídeo puede determinar si un píxel de una imagen de los datos de vídeo es un píxel de escape. El píxel puede ser un píxel de escape cuando un valor de muestra del píxel corresponda a un valor de muestra especificado por una entrada de paleta en una paleta que comprenda entradas de paleta que especifiquen valores de muestra. En respuesta a la determinación de que el píxel no es un píxel escapado, el
10 codificador de vídeo puede determinar un índice para el píxel e incluir, en un flujo de bits que comprenda una representación codificada de los datos de vídeo, datos que indiquen el índice para el píxel. En respuesta a la determinación de que el píxel es un píxel de escape, el codificador de vídeo puede binarizar el valor de muestra del píxel para generar una contraseña de longitud fija. Además, como respuesta a la determinación de que el píxel es un píxel escapado, el codificador de vídeo puede usar CABAC con modelado de contexto para codificar por entropía la primera casilla de la contraseña de longitud fija. Además, en respuesta a la determinación de que el píxel en un píxel
15 escapado, el codificador de vídeo puede usar la codificación de omisión CABAC para codificar por entropía cada bin de la contraseña de longitud fija que sigue al primer bin de la contraseña de longitud fija. En algunos ejemplos, la longitud de la contraseña de longitud fija depende de los valores de QP para cada canal (por ejemplo, luminancia, crominancia, etc. canal) de la imagen.

20 **[0124]** Ahora se describe una técnica de ejemplo de omisión de límite de error de paleta. En algunos ejemplos, el tamaño de la paleta está relacionado con QP. Por ejemplo, se puede asignar un límite de error de paleta mayor para un QP más grande, por tanto grupos más pequeños de índices de paleta; se puede asignar un límite de error de paleta más pequeño para QP más pequeños, por tanto, más grupos de índices de paleta. Se puede usar una tabla de asignación (tabla de consulta) de 52 entradas en la memoria para almacenar la relación entre cada valor de QP y el
25 límite de error de paleta.

[0125] Por ejemplo, en algunos ejemplos, un decodificador de vídeo puede determinar un índice para un píxel de una imagen de los datos de vídeo. Además, el decodificador de vídeo puede determinar, basándose en el índice para el píxel, una entrada en una paleta, en el que la entrada determinada especifica un valor de muestra para el píxel, en el que un tamaño de la paleta está relacionado con un QP. De forma similar, en algunos ejemplos, un codificador de vídeo puede incluir, en un flujo de bits que comprenda una representación codificada de los datos de vídeo, datos que indiquen un índice de una entrada en una paleta, en el que un tamaño de la paleta está relacionado con un parámetro de cuantización. En algunos de estos ejemplos, el codificador de vídeo puede determinar, basándose en el parámetro de cuantización, un límite de error de paleta. En dichos ejemplos, el codificador de vídeo puede incluir, en el flujo de bits, los datos que indiquen el índice de la entrada en la paleta solo si la diferencia entre un valor de muestra especificado por la entrada y un valor de muestra del píxel es menor que el límite de error de paleta.
30

[0126] Ahora se describe una técnica de ejemplo de derivación de píxeles de escape cuantizada. En algunos ejemplos, el parámetro de cuantización para un píxel de escape (o error de predicción) de cada canal es el mismo que el parámetro de cuantización para la codificación de coeficientes tradicionales. En otras palabras, la cuantización o la descuantización de píxeles de escape (error de predicción) pueden ser diferentes en diferentes canales. En un codificador de vídeo, cada canal del píxel de escape usa el parámetro de cuantización para la codificación de coeficiente tradicional. En un decodificador de vídeo, cada canal del píxel de escape usa el parámetro de cuantización recibido para la codificación de coeficiente tradicional para reconstruir el valor de píxel de escape o el error de predicción de píxel de escape.
35
40
45

[0127] En algunos ejemplos, un decodificador de vídeo puede determinar si un píxel de una imagen de los datos de vídeo es un píxel de escape. En respuesta a la determinación de que el píxel no es un píxel escapado, el decodificador de vídeo puede determinar, basándose en un índice para el píxel, una entrada en una paleta que comprenda entradas que especifiquen valores de muestra, especificando la entrada determinada un valor de muestra del píxel. En respuesta a la determinación de que el píxel es un píxel de escape, el decodificador de vídeo puede determinar, basándose en uno o más elementos sintácticos en el flujo de bits y sin determinar una entrada en la paleta, el valor de muestra del píxel. Además, respondiendo a la determinación de que el píxel es un píxel escapado, el decodificador de vídeo puede descuantizar el valor de muestra del píxel. El valor de muestra del píxel puede ser un valor de muestra residual del píxel, y el decodificador de vídeo puede agregar un valor de muestra predictivo para el píxel al valor de muestra residual del píxel para determinar un valor de muestra decodificado para el píxel. Además, en algunos de estos ejemplos, el valor de muestra del píxel es un primer valor de muestra del píxel, el parámetro de cuantización es un primer parámetro de cuantización, el primer valor de muestra del píxel y el primer parámetro de cuantización corresponden a un primer canal. En dichos ejemplos, en respuesta a la determinación de que el píxel es un píxel escapado, la decodificación de vídeo puede descuantizarse, basándose en un segundo parámetro de cuantización, en un segundo valor de muestra del píxel, en el segundo valor de muestra del píxel y en el segundo parámetro de cuantización correspondiente al segundo canal.
50
55
60

[0128] En un ejemplo similar, un codificador de vídeo puede determinar si un píxel de una imagen de los datos de vídeo es un píxel de escape. El píxel puede ser un píxel de escape cuando un valor de muestra del píxel no corresponda a una entrada en una paleta que comprenda entradas que especifiquen valores de muestra. En respuesta
65

a la determinación de que el píxel es un píxel escapado, el codificador de vídeo puede cuantizar, basándose en un parámetro de cuantización, el valor de muestra del píxel e incluir, en un flujo de bits que comprenda una representación codificada de los datos de vídeo, datos que indiquen el valor de muestra cuantizado del píxel. En respuesta a la determinación de que el píxel no es un píxel escapado, el codificador de vídeo puede determinar una entrada en la paleta que corresponda al valor de muestra del píxel e incluir, en el flujo de bits, datos que indiquen un índice de la entrada determinada en la paleta. En algunos ejemplos, el valor de muestra para el píxel es un valor de muestra residual del píxel. Además, en algunos ejemplos, el valor de muestra del píxel es un primer valor de muestra del píxel, el parámetro de cuantización es un primer parámetro de cuantización, el primer valor de muestra del píxel y el primer parámetro de cuantización corresponden a un primer canal y responde para determinar que el píxel es un píxel escapado, el codificador de vídeo puede cuantizar, basándose en un segundo parámetro de cuantización, un segundo valor de muestra del píxel, el segundo valor de muestra del píxel y el segundo parámetro de cuantización correspondiente al segundo canal. El codificador de vídeo puede incluir, en el flujo de bits, datos que indiquen el segundo valor de muestra cuantizado del píxel.

[0129] La FIG. 2 es un diagrama de bloques que ilustra un codificador de vídeo 20 de ejemplo que puede implementar diversas técnicas de esta divulgación. La FIG. 2 se proporciona para propósitos de explicación y no debería considerarse limitadora de las técnicas tales como las ampliamente ejemplificadas y descritas en esta divulgación. Para los propósitos de explicación, esta divulgación describe el codificador de vídeo 20 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

[0130] En el ejemplo de la FIG. 2, el codificador de vídeo 20 incluye una memoria de datos de vídeo 98, una unidad de procesamiento de predicción 100, una unidad de generación residual 102, una unidad de procesamiento de transformada 104, una unidad de cuantización 106, una unidad de cuantización 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 decodificadas 116 y una unidad de codificación por entropía 118. La unidad de procesamiento de predicción 100 incluye una unidad de procesamiento de interpredicción 120 y una unidad de procesamiento de intrapredicción 126. La unidad de procesamiento de interpredicción 120 incluye una unidad de estimación de movimiento y una unidad de compensación de movimiento (no mostrada). El codificador de vídeo 20 también incluye una unidad de codificación basada en paletas 122, configurada para realizar diversos aspectos de las técnicas de codificación basada en paletas descritas en esta divulgación. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0131] La memoria de datos de vídeo 98 puede almacenar datos de vídeo para codificarse por los componentes del codificador de vídeo 20. Los datos de vídeo almacenados en la memoria de datos de vídeo 98 se pueden obtener, por ejemplo, a partir de la fuente de vídeo 18. La memoria intermedia de imágenes decodificadas 116 puede ser una memoria de imágenes de referencia que almacene datos de vídeo de referencia para su uso en la codificación de datos de vídeo mediante el codificador de vídeo 20, por ejemplo, en modalidades de intracodificación o intercodificación. La memoria de datos de vídeo 98 y la memoria intermedia de imágenes decodificadas 116 pueden estar formadas por cualquiera entre varios dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 98 y la memoria intermedia de imágenes decodificadas 116 pueden proporcionarse mediante el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 98 puede estar en un chip con otros componentes del codificador de vídeo 20, o fuera de chip con respecto a esos componentes.

[0132] El codificador de vídeo 20 puede recibir datos de vídeo. El codificador de vídeo 20 puede codificar cada CTU en un fragmento de una imagen de los datos de vídeo. Cada una de las CTU puede asociarse a bloques de árbol de codificación (CTB) de luma de igual tamaño y a los CTB correspondientes de la imagen. Como parte de la codificación de una CTU, la unidad de procesamiento de predicción 100 puede realizar la partición de árbol cuádruple para dividir los CTB de la CTU en bloques progresivamente más pequeños. Los bloques más pequeños pueden ser bloques de codificación de CU. Por ejemplo, la unidad de procesamiento de predicción 100 puede dividir un CTB asociado a una CTU en cuatro subbloques de igual tamaño, dividir uno o más de los subbloques en cuatro subsubbloques de igual tamaño, etc.

[0133] 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 particionar los bloques de codificación asociados a la CU entre una o más PU de la CU. Por lo tanto, cada PU puede asociarse a un bloque predictivo de luma y a los correspondientes bloques predictivos de croma. El codificador de vídeo 20 y el decodificador de vídeo 30 pueden dar soporte a las PU de diversos tamaños. Como se ha indicado anteriormente, el tamaño de una CU puede referirse al tamaño del bloque de codificación de luma de la CU, y el tamaño de una PU puede referirse al tamaño de un bloque predictivo de luma de la PU. Suponiendo que el tamaño de una CU particular sea de $2N_x \times 2N_y$, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden dar soporte a tamaños de PU de $2N_x \times 2N_y$ o $N_x \times N_y$ para la intrapredicción, y a tamaños de PU simétricas de $2N_x \times 2N_y$, $2N_x \times N_y$, $N_x \times 2N_y$ o similares para la interpredicción. El codificador de vídeo 20 y el decodificador de vídeo 30 también pueden dar soporte a la partición asimétrica para tamaños de PU de $2N_x \times nU$, $2N_x \times nD$, $nL \times 2N_y$ y $nR \times 2N_y$, para la interpredicción.

[0134] La unidad de procesamiento de interpredicción 120 puede generar datos predictivos para una PU realizando una interpredicción en cada PU de una CU. Los datos predictivos para la PU pueden incluir uno o más bloques de muestras predictivas de la PU y la información de movimiento para la PU. La unidad de interpredicción 121 puede realizar diferentes operaciones para una PU de una CU dependiendo de si la PU está en un fragmento I, en un fragmento P o en un fragmento B. En un fragmento en I, todas las PU se intrapredicen. Por lo tanto, si la PU está en un fragmento I, la unidad de interpredicción 121 no realiza interpredicción en la PU. Por tanto, para bloques codificados en el modo I, el bloque predictivo se forma usando predicción espacial a partir de bloques adyacentes previamente codificados dentro de la misma trama.

[0135] Si una PU está en un fragmento P, la unidad de estimación de movimiento de la unidad de procesamiento de interpredicción 120 puede buscar las imágenes de referencia en una lista de imágenes de referencia (por ejemplo, "ListalmagenRef0") para una región de referencia para la PU. La región de referencia para la PU puede ser una región, dentro de una imagen de referencia, que contenga bloques de muestra que se correspondan más estrechamente con los bloques de muestra de la PU. La unidad de estimación de movimiento 122 puede generar un índice de referencia que indique una posición en la ListalmagenRef0 de la imagen de referencia que contenga la región de referencia para la PU. Además, la unidad de estimación de movimiento puede generar un vector de movimiento que indique un desplazamiento espacial entre un bloque de codificación de la PU y una ubicación de referencia asociada a la región de referencia. Por ejemplo, el vector de movimiento puede ser un vector bidimensional que proporcione un desplazamiento entre las coordenadas en la imagen decodificada actual y las coordenadas en una imagen de referencia. La unidad de estimación de movimiento puede emitir el índice de referencia y el vector de movimiento como la información de movimiento de la PU. La unidad de compensación de movimiento de la unidad de procesamiento de interpredicción 120 puede generar los bloques predictivos de la PU basándose en muestras reales o interpoladas en la ubicación de referencia, indicada por el vector de movimiento de la PU.

[0136] Si una PU está en un fragmento B, la unidad de estimación de movimiento puede realizar unipredicción o bipredicción para la PU. Para realizar la unipredicción para la PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia de la ListalmagenRef0 o de una segunda lista de imágenes de referencia ("ListalmagenRef1") para una región de referencia para la PU. La unidad de estimación de movimiento puede emitir, como la información de movimiento de la PU, un índice de referencia que indique una posición en la ListalmagenRef0 o en la ListalmagenRef1 de la imagen de referencia que contiene la región de referencia, un vector de movimiento que indique un desplazamiento espacial entre un bloque predictivo de la PU y una ubicación de referencia asociada a la región de referencia, y uno o más indicadores de dirección de predicción que indiquen si la imagen de referencia está en la ListalmagenRef0 o en la ListalmagenRef1. La unidad de compensación de movimiento de la unidad de procesamiento de interpredicción 120 puede generar los bloques predictivos de la PU basándose, al menos en parte, en muestras reales o interpoladas en la región de referencia indicada por el vector de movimiento de la PU.

[0137] Para realizar la interpredicción bidireccional para una PU, la unidad de estimación de movimiento puede buscar las imágenes de referencia en la ListalmagenRef0 para una región de referencia para la PU y también puede buscar las imágenes de referencia en la ListalmagenRef1 para otra región de referencia para la PU. La unidad de estimación de movimiento puede generar índices de imágenes de referencia que indiquen posiciones en la ListalmagenRef0 y en la ListalmagenRef1 de las imágenes de referencia que contienen las regiones de referencia. Además, la unidad de estimación de movimiento puede generar vectores de movimiento que indiquen desplazamientos espaciales entre la ubicación de referencia asociada a las regiones de referencia y un bloque de muestra de la PU. La información de movimiento de la PU puede incluir los índices de referencia y los MV de la PU. La unidad de compensación de movimiento puede generar los bloques predictivos de muestra de la PU basándose, al menos en parte, en muestras reales o interpoladas en las regiones de referencia indicadas por los vectores de movimiento de la PU.

[0138] De acuerdo con diversos ejemplos de esta divulgación, el codificador de vídeo 20 puede configurarse para realizar una codificación basada en paletas. Con respecto al marco de HEVC, como ejemplo, las técnicas de codificación basada en paletas pueden configurarse para usarse como un modo CU. En otros ejemplos, las técnicas de codificación basada en paletas pueden configurarse para usarse como una modalidad de PU en el entorno de la HEVC. Por consiguiente, todos los procesos divulgados, descritos en el presente documento (a lo largo de esta divulgación) en el contexto de una modalidad de CU, pueden, adicionalmente o alternativamente, aplicarse a las PU. Sin embargo, estos ejemplos basados en la HEVC no deberían considerarse como una restricción o limitación de las técnicas de codificación basada en paletas, descritas en el presente documento, ya que dichas técnicas se pueden aplicar para que funcionen independientemente o como parte de otros sistemas/normas existentes o por desarrollar. En estos casos, la unidad para la codificación por paleta pueden ser bloques cuadrados, bloques rectangulares o incluso regiones de forma no rectangular.

[0139] La unidad de codificación basada en paletas 122, por ejemplo, puede realizar una decodificación basada en paletas cuando se seleccione una modalidad de codificación basada en paletas, por ejemplo, para una CU o PU. Por ejemplo, la unidad de codificación basada en paletas 122 puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxeles, seleccionar valores de píxeles en una paleta para representar valores de píxeles de al menos algunas posiciones de un bloque de datos de vídeo y señalar información que asocie al menos

algunas de las posiciones del bloque de datos de vídeo con entradas en la paleta correspondientes, respectivamente, a los valores de píxeles seleccionados. Aunque se describen diversas funciones como realizadas por la unidad de codificación basada en paletas 122, algunas de, o todas, estas funciones pueden realizarse por otras unidades de procesamiento, o una combinación de diferentes unidades de procesamiento.

5 **[0140]** La unidad de codificación basada en paletas 122 puede estar configurada para generar cualquiera de los diversos elementos sintácticos descritos en el presente documento. Por consiguiente, el codificador de vídeo 20 puede configurarse para codificar bloques de datos de vídeo usando modos de código basados en paletas como se describe en esta divulgación. El codificador de vídeo 20 puede codificar selectivamente un bloque de datos de vídeo usando un modo de codificación por paleta, o codificar un bloque de datos de vídeo usando un modo diferente, por ejemplo, un modo de codificación HEVC interpredictiva o intrapredictiva. El bloque de datos de vídeo puede ser, por ejemplo, una CU o PU generada de acuerdo con un proceso de codificación de HEVC. Un codificador de vídeo 20 puede codificar algunos bloques con modos de predicción temporal interpredictiva o codificación espacial intrapredictiva y decodificar otros bloques con el modo de codificación basado en paleta.

15 **[0141]** La unidad de procesamiento de intrapredicción 126 puede generar datos predictivos para una PU realizando una intrapredicción en la PU. Los datos predictivos para la PU pueden incluir bloques predictivos para la PU y diversos elementos sintácticos. La unidad de procesamiento de intrapredicción 126 puede realizar la intrapredicción en las PU en fragmentos I, fragmentos P y fragmentos B.

20 **[0142]** Para realizar la intrapredicción en una PU, la unidad de procesamiento de intrapredicción 126 puede usar múltiples modalidades de intrapredicción para generar múltiples conjuntos de datos predictivos para la PU. Cuando se usan algunos modos de intrapredicción para generar un conjunto de datos predictivos para la PU, la unidad de procesamiento de intrapredicción 126 puede extender los valores de las muestras de bloques de muestra de las PU adyacentes a través de los bloques predictivos de la PU en direcciones asociadas con los modos de intrapredicción. Las PU adyacentes pueden estar arriba, arriba y a la derecha, arriba y a la izquierda, o a la izquierda de la PU, suponiendo un orden de codificación de izquierda a derecha, de arriba a abajo, para las PU, CU y CTU. La unidad de procesamiento de intrapredicción 126 puede usar diversos números de modalidades de intrapredicción, por ejemplo, 33 modalidades de intrapredicción direccional. En algunos ejemplos, el número de modalidades de intrapredicción puede depender del tamaño de la región asociada a la PU.

25 **[0143]** La unidad de procesamiento de predicción 100 puede seleccionar los datos predictivos para las PU de una CU a partir de los datos predictivos generados por la unidad de procesamiento de interpredicción 120 para las PU o los datos predictivos generados por la unidad de procesamiento de intrapredicción 126 para las PU. En algunos ejemplos, la unidad de procesamiento de predicción 100 selecciona los datos predictivos para las PU de la CU basándose en las mediciones de velocidad/distorsión de los conjuntos de datos predictivos. Los bloques de muestras predictivas de los datos predictivos seleccionados pueden denominarse en el presente documento como los bloques predictivos de muestra seleccionados.

35 **[0144]** La unidad de generación residual 102 puede generar, basándose en los bloques de codificación (por ejemplo, los bloques de codificación de Luma, Cb y Cr) de una CU y los bloques predictivos de muestra seleccionados (por ejemplo, los bloques predictivos de Luma, Cb y Cr seleccionados) de las PU de la CU, bloques residuales (por ejemplo, bloques residuales de luma, Cb y Cr) de la CU. Por ejemplo, la unidad de generación residual 102 puede generar los bloques residuales de la CU de modo que cada muestra en los bloques residuales tenga un valor igual a una diferencia entre una muestra en un bloque de codificación de la CU y una muestra correspondiente en un correspondiente bloque predictivo seleccionado de una PU de la CU.

40 **[0145]** La unidad de procesamiento de transformada 104 puede realizar la partición de árbol cuádruple para particionar los bloques residuales de una CU en bloques de transformada asociados a las TU de la CU. Por tanto, en algunos ejemplos, una TU puede asociarse con un bloque de transformada de luma y dos bloques de transformada de croma. Los tamaños y las posiciones de los bloques de transformada de luma y croma de las TU de una CU pueden o no basarse en los tamaños y posiciones de los bloques predictivos 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.

50 **[0146]** La unidad de procesamiento de transformada 104 puede generar bloques de coeficientes de transformada para cada TU de una CU aplicando una o más transformadas a los bloques de transformada de la TU. La unidad de procesamiento de transformada 104 puede aplicar diversas transformadas a un bloque de transformada asociado a una TU. Por ejemplo, la unidad de procesamiento de transformada 104 puede aplicar una transformada de coseno discreta (DCT), una transformada direccional o una transformada conceptualmente similar al bloque de transformada. En algunos ejemplos, la unidad de procesamiento de transformada 104 no aplica transformadas a un bloque de transformada. En dichos ejemplos, el bloque de transformada puede tratarse como un bloque de coeficientes de transformada.

60 **[0147]** La unidad de cuantización 106 puede cuantizar los coeficientes de transformada en un bloque de coeficientes. El proceso de cuantización puede reducir la profundidad de bits asociada a algunos o a la totalidad de los coeficientes

65

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

[0148] La unidad de cuantización inversa 108 y la unidad de procesamiento de transformada inversa 110 pueden aplicar la cuantización inversa y transformadas inversas al bloque de coeficientes, respectivamente, para reconstruir un bloque residual a partir del bloque de coeficientes. La unidad de reconstrucción 112 puede añadir el bloque residual reconstruido a las muestras correspondientes de uno o más bloques predictivos generados por la unidad de procesamiento de predicción 100 para producir un bloque de transformada reconstruido asociado con una TU. Mediante la reconstrucción de bloques de transformada para cada TU de una CU de esta manera, el codificador de vídeo 20 puede reconstruir los bloques de codificación de la CU.

[0149] La unidad de filtro 114 puede realizar una o más operaciones de desbloqueo para reducir las distorsiones de bloqueo en los bloques de codificación asociados con una CU. La memoria intermedia de imágenes decodificadas 116 puede almacenar los bloques de codificación reconstruidos después de que la unidad de filtro 114 realice las una o más operaciones de desbloqueo en los bloques de codificación reconstruidos. La unidad de procesamiento de interpredicción 120 puede usar una imagen de referencia que contenga los bloques de codificación reconstruidos para realizar la interpredicción en las PU de otras imágenes. Además, la unidad de procesamiento de intrapredicción 126 puede usar bloques de codificación reconstruidos en la memoria intermedia de imágenes decodificadas 116 para realizar intrapredicción en otras PU en la misma imagen que la CU.

[0150] La unidad de codificación por entropía 118 puede recibir datos desde otros componentes funcionales del codificador de vídeo 20. Por ejemplo, la unidad de codificación por entropía 118 puede recibir bloques de coeficientes desde la unidad de cuantización 106 y puede recibir elementos sintácticos desde la unidad de procesamiento de predicción 100. La unidad de codificación por entropía 118 puede realizar una o más operaciones de codificación por entropía en los datos para generar datos codificados por entropía. Por ejemplo, la unidad de codificación por entropía 118 puede realizar una operación de CABAC, una operación de codificación de longitud variable adaptativa al contexto (CAVLC), una operación de codificación de longitud variable a variable (V2V), una operación de codificación aritmética binaria adaptativa al contexto basada en la sintaxis (SBAC), una operación de codificación por entropía de partición de intervalos de probabilidad (PIPE), una operación de codificación exponencial de Golomb u otro tipo de operación de codificación por entropía en los datos. El codificador de vídeo 20 puede emitir un flujo de bits que incluya datos codificados por entropía, generados por la unidad de codificación por entropía 118. Por ejemplo, el flujo de bits puede incluir datos que representen un RQT para una CU.

[0151] En algunos ejemplos, la codificación residual no se realiza con la codificación de paleta. Por consiguiente, el codificador de vídeo 20 no puede realizar la transformada o la cuantización cuando se codifique usando un modo de codificación de paleta. Además, el codificador de vídeo 20 puede codificar por entropía los datos generados usando un modo de codificación de paleta independientemente de los datos residuales.

[0152] De acuerdo con una o más de las técnicas de esta divulgación, el codificador de vídeo 20, y específicamente la unidad de codificación basada en paletas 122, puede realizar la codificación de vídeo basada en paletas de bloques de vídeo predichos. Como se describió anteriormente, una paleta generada por el codificador de vídeo 20 puede codificarse explícitamente y enviarse al decodificador de vídeo 30, predecirse a partir de entradas de paleta anteriores, predecirse a partir de valores de píxeles anteriores, o una combinación de los mismos.

[0153] La unidad de codificación basada en paletas 122 puede aplicar técnicas de esta divulgación para derivar la codificación de un mapa de valores de índice de paleta para los píxeles de un bloque codificado por paleta en ciertas circunstancias. De acuerdo con los aspectos de esta divulgación, la unidad de codificación basada en paletas 122 puede omitir la codificación del mapa de valores de índice de paleta para un bloque codificado por paleta si la unidad de codificación basada en paleta 122 determina que todos los píxeles del bloque son del mismo color. Por ejemplo, la unidad de codificación basada en paletas 122 puede determinar que una CU codificada por paleta de una imagen es una "CU de un único color" y puede omitir la codificación y la señalización del mapa de valores de índice de paleta para la CU de un único color.

[0154] Más específicamente, la unidad de codificación basada en paletas 122 puede determinar si una CU codificada por paleta es una CU de un único color si la CU actual cumple con dos condiciones. La primera condición que la unidad de codificación basada en paletas 122 puede usar para determinar si la CU codificada por paleta es una CU de un único color es si el tamaño de la paleta correspondiente es igual a uno (1). Si el tamaño de la paleta es igual a uno, entonces la unidad de codificación basada en paletas 122 puede determinar que la primera condición se cumple con respecto a la CU codificada por paleta que es una CU de un único color. Más específicamente, si el tamaño de la paleta es uno, entonces la unidad de codificación basada en paletas 122 puede determinar que la paleta incluye solo un color que corresponde a los píxeles (no de escape) de la CU codificada por paleta. En algunos ejemplos donde el

tamaño de la paleta es uno, la unidad de codificación basada en paletas 122 puede determinar que el único valor de índice incluido en la paleta es cero (0).

5 **[0155]** Si la unidad de codificación basada en paletas 122 determina que el tamaño de la paleta para la CU codificada por paleta es 1 (es decir, que se cumple con la primera condición), entonces la unidad de codificación basada en paletas 122 puede determinar si la CU codificada por paleta cumple con una segunda condición de ser una CU de un único color. La segunda condición que la unidad de codificación basada en paletas 122 puede usar para determinar si la CU codificada por paleta es una CU de un único color es que la CU codificada por paleta no incluye ningún píxel de escape. Si la CU codificada por paleta incluye al menos un píxel de escape, el codificador de vídeo 20 puede determinar que, aunque la paleta correspondiente solo indica un color con respecto a la CU codificada por paleta, la CU codificada por paleta incluye píxeles de dos o más colores. Por ejemplo, la CU codificada por paleta puede incluir al menos un píxel que tenga el color indicado en la paleta, y al menos un píxel de escape que tenga un color diferente.

15 **[0156]** Si la unidad de codificación basada en paletas 122 determina que la CU codificada por paleta cumple con las dos condiciones descritas anteriormente, entonces la unidad de codificación basada en paletas 122 puede determinar que la CU codificada por paleta es una CU de un único color. Más específicamente, si la CU codificada por paleta se asocia con una paleta de entrada única (mostrada por el tamaño de la paleta de uno), y la CU codificada por paleta no incluye ningún píxel de escape, entonces la unidad de codificación basada en paletas 122 puede determinar que todos los píxeles individuales de la CU codificada por paleta son del mismo color (es decir, el color indicado por la entrada única de la paleta correspondiente). En diversas implementaciones, la unidad de codificación basada en paletas 122 puede aplicar las técnicas de identificación de CU de un único color descritas anteriormente con respecto a una base de componente de un único color con respecto a la paleta, o a un índice combinado que indica más de un componente de color.

25 **[0157]** En respuesta a determinar que la CU codificada por paleta incluye píxeles de un único color (es decir, que la CU es una CU de un único color), la unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para evitar la codificación del mapa de valores de índice de paleta para los píxeles de la CU de un único color. Al omitir la codificación del mapa de valores de índice de paleta para la CU, la unidad de codificación basada en paletas 122 puede conservar los recursos informáticos y el ancho de banda que de otra manera se utilizaría para codificar y señalar la información de color para la CU. En lugar de codificar un mapa de valores de índice de paleta para los píxeles de la CU de un único color, la unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para indicar de manera más eficiente la información de color para el conjunto completo de píxeles que forman la CU de un único color. En los casos en que la unidad de codificación basada en paletas 122 derive la codificación del índice de paleta para todos los píxeles individuales de una CU de un único color detectado, el codificador de vídeo 20 (o uno o más componentes del mismo) puede no necesitar señalar un mapa de los valores del índice de la paleta para los píxeles de la CU, lo que conserva los recursos informáticos y reduce el consumo de ancho de banda.

40 **[0158]** De acuerdo con algunos ejemplos de esta divulgación, la unidad de codificación basada en paletas 122 puede codificar un indicador para indicar que el codificador de vídeo 20 omitió la codificación (y señalización) del mapa de valores de índice de paleta línea por línea para la CU de un único color. Al codificar un indicador de un bit para la CU completa en lugar del mapa (de cualquier tamaño) de valores de índice de paleta para los píxeles de la CU, la unidad de codificación basada en paletas 122 puede causar que el codificador de vídeo 20 conserve los recursos informáticos y el ancho de banda de señalización en comparación con las técnicas de codificación basada en paletas existentes. Además, la unidad de codificación basada en paletas 122 puede mantener la precisión y la calidad de la CU codificada de un único color, porque la paleta de entrada única codificada por la unidad de codificación basada en paletas 122 y señalizada por el codificador de vídeo 20 para la CU incluye la información de color para todos los píxeles de la CU. En diversos ejemplos, la unidad de codificación basada en paletas 122 puede codificar el indicador de diversas formas, como en un SPS, un PPS o un encabezado de fragmento. En diversos ejemplos, la unidad de codificación basada en paletas 122 puede codificar el indicador por CTU, por CU o también para un bloque de cualquier tamaño de bloque.

55 **[0159]** En otros ejemplos, la unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para derivar un límite de error para una paleta para un bloque codificado por paleta. Como se usa en el presente documento, los términos "límite de error" o "límite de error de paleta" pueden referirse a la variación mínima de los valores de píxeles (por ejemplo, en términos de información de color) que pueden incluir las entradas de la paleta. Más específicamente, el límite de error de paleta define una variación mínima en el valor del color que deben mostrar cualquiera de las dos entradas de la paleta. Como se describe, para codificar un bloque de acuerdo con la codificación basada en paletas, la unidad de codificación basada en paletas 122 puede construir la paleta correspondiente para incluir valores de color que se produzcan con mayor frecuencia (píxel por píxel) dentro del bloque.

60 **[0160]** En otras palabras, la unidad de codificación basada en paletas 122 puede construir una paleta agrupando los valores de píxeles de alta ocurrencia (u ocurrencia relativamente más alta) del bloque en la paleta. La unidad de codificación basada en paletas 122 puede seleccionar los valores de píxeles de alta ocurrencia de modo que los valores de los píxeles muestren al menos una variación mínima particular, en términos de información de color. A su vez, la variación mínima del valor de píxel dentro del conjunto seleccionado de valores de píxeles de alta ocurrencia puede formar el límite de error de la paleta correspondiente. Se apreciará que, aunque el límite de error de paleta

pueda incluir varios valores de píxeles, la paleta puede no incluir necesariamente cada par de valores de píxeles que muestren al menos el límite de error de paleta. Por lo tanto, el mismo límite de error de paleta puede aplicarse a paletas de diferentes tamaños. La unidad de codificación basada en paletas 122 puede usar el límite de error de paleta para determinar los valores de color que se incluirán en la paleta.

[0161] La unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para definir el límite de error para una paleta. De acuerdo con diversos aspectos de esta divulgación, la unidad de codificación basada en paletas 122 puede determinar el límite de error de paleta basándose en el parámetro de cuantización (QP) para el bloque codificado por paleta. En diversos ejemplos, la unidad de codificación basada en paletas 122 puede determinar que el límite de error de paleta es directamente proporcional al valor QP para el bloque correspondiente. Más específicamente, en estos ejemplos, la unidad de codificación basada en paletas puede asignar un límite de error mayor para una paleta para un bloque que se cuantice con un valor de QP mayor y un límite de error menor para una paleta para un bloque que se cuantice con un valor QP menor.

[0162] Adicionalmente, la unidad de codificación basada en paletas 122 puede implementar las técnicas descritas en el presente documento para derivar el límite de error de paleta almacenando una tabla que asigne el valor QP de un bloque codificado por paleta al límite de error de paleta correspondiente. A su vez, la unidad de codificación basada en paletas 122 puede acceder a una entrada de límite de error de paleta particular desde la tabla para determinar la paleta correspondiente. Al usar una tabla que asigne el límite de error de la paleta al QP de un bloque codificado por paleta, la unidad de codificación basada en paletas 122 puede mejorar la eficiencia computacional en comparación con las técnicas de codificación existentes basadas en paletas. Más específicamente, al implementar las técnicas de derivación del límite de error de la paleta basada en la tabla de asignación descritas en el presente documento, la unidad de codificación basada en paletas 122 puede conservar recursos que de otra manera se utilizarían de acuerdo con las técnicas existentes, que se basan en la ejecución de una función para derivar el límite de error para cada paleta.

[0163] Por lo tanto, la unidad de codificación basada en paletas 122 puede definir paletas que pueden acomodar un mayor número de índices de paleta para requerir una mayor variación entre los valores de píxeles para los bloques que se cuantizan con valores de QP mayores, y puede definir paletas que requieran una variación menor entre los valores de píxeles que se cuantizan con mayores valores de QP. De esta manera, la unidad de codificación basada en paletas 122 puede implementar diversas técnicas de esta divulgación para personalizar la paleta (basándose en el límite de error de paleta) basándose en el valor QP con el que se cuantiza el bloque correspondiente, determinando de este modo el contenido de la paleta para un bloque basado en el valor QP del bloque. Adicionalmente, la unidad de codificación basada en paletas 122 puede generar y/o almacenar una tabla (por ejemplo, una tabla de asignación o tabla de consulta) para reflejar la relación entre cada valor de QP y el límite de error de paleta correspondiente. De esta manera, la unidad de codificación basada en paletas 122 puede implementar las técnicas de esta divulgación para derivar un límite de error de paleta para una paleta correspondiente de una manera menos costosa desde el punto de vista computacional que las técnicas de codificación de paleta existentes, que se basan en resolver una función para derivar el límite de error para cada paleta.

[0164] La unidad de codificación basada en paletas 122 puede, en algunos ejemplos, implementar diversas técnicas de esta divulgación para la derivación cuantizada de píxeles de escape. Más específicamente, la unidad de codificación basada en paletas 122 puede implementar las técnicas para definir el valor de cuantización del QP para un píxel de escape. Por ejemplo, de acuerdo con las técnicas de codificación basada en paletas, si la unidad de codificación basada en paletas 122 detecta un píxel de escape en un bloque codificado por paleta, la unidad de codificación basada en paletas 122 puede codificar el valor del píxel, o un error de predicción del mismo, debido a que la paleta no incluye ninguna entrada para el píxel de escape. Adicionalmente, para conservar el ancho de banda de señalización, la unidad de cuantización 106 puede cuantizar el valor de píxel codificado del píxel de escape antes de la señalización de otros componentes del codificador de vídeo 20.

[0165] De acuerdo con las técnicas de codificación basada en paletas existentes, no se definió ningún valor de cuantización (valor de QP) para cuantizar un píxel de escape. La unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para definir el valor de QP para cuantizar un píxel de escape. Más específicamente, la unidad de codificación basada en paletas 122 puede definir el valor de QP para un píxel de escape igual al valor de QP para la codificación de coeficiente tradicional dentro del mismo canal de color. Por tanto, la unidad de codificación basada en paletas 122 puede cuantizar todos los píxeles de escape de acuerdo con un único valor de QP, dentro de un canal de color dado. Adicionalmente, como la unidad de codificación basada en paletas 122 puede definir el valor QP para todos los píxeles de escape solo dentro de un único canal de color, la unidad de codificación basada en paletas 122 puede usar diferentes valores de QP para cuantizar los píxeles de escape con respecto a diferentes canales de color.

[0166] Además, la unidad de codificación basada en paletas 122 puede implementar otras técnicas de esta divulgación para usar un indicador para indicar y/o detectar la inclusión de un píxel de escape en un bloque codificado por paleta. De acuerdo con las técnicas de codificación basada en paletas existentes, los píxeles de escape pueden señalizarse y detectarse usando un valor de índice de paleta "reservado". Por ejemplo, de acuerdo con las técnicas de codificación basada en paletas existentes, el valor de índice de paleta reservado que indica un píxel de escape

puede ser 32. Más específicamente, el valor de índice de la paleta de 32 se puede usar para todos los píxeles de escape, independientemente de si dos píxeles de escape tienen valores de píxeles diferentes. Por tanto, de acuerdo con las técnicas de codificación basada en paletas existentes, los dispositivos de codificación de vídeo pueden usar un valor de cinco bits (de 32) para cada píxel de escape de un bloque codificado por paleta.

5
 [0167] La unidad de codificación basada en paletas 122 puede implementar técnicas de esta divulgación para conservar los recursos informáticos (por ejemplo, el almacenamiento y la memoria) y reducir el consumo de ancho de banda, mientras mantiene la precisión de la imagen con respecto a la señalización de un píxel de escape en un bloque codificado por paleta. Por ejemplo, la unidad de codificación basada en paletas 122 puede codificar un indicador (por ejemplo, una unidad de datos de un bit) para indicar si un píxel en un bloque codificado por paleta es un píxel de escape. Como se describe en el presente documento, el indicador, cuando está habilitado, puede indicar que al píxel asociado se le asigna un índice de paleta denominado "otro índice". La unidad de codificación basada en paletas 122 puede usar el estado de "otro índice" del indicador para reemplazar el valor de índice de paleta de 32 que tradicionalmente se usa para indicar un píxel de escape con respecto a la paleta. Por tanto, la unidad de codificación basada en paletas 122 puede codificar (y otros componentes del codificador de vídeo 20 pueden indicar) un indicador de un bit en lugar de un valor de índice de cinco bits para indicar que un píxel de un bloque codificado por paleta es un píxel de escape. A su vez, el codificador de vídeo 20 (o uno o más componentes del mismo) puede codificar y señalar el valor de píxel, o sus datos residuales, en el flujo de bits de vídeo codificado.

20
 [0168] Una o ambas de la unidad de codificación basada en paletas 122 y de la unidad de cuantización 106 también pueden implementar técnicas de esta divulgación para cuantizar los valores de píxeles de los píxeles de escape de acuerdo con la codificación basada en paletas. Por ejemplo, la unidad de codificación basada en paletas 122 y/o la unidad de cuantización 106 pueden conservar los recursos informáticos (por ejemplo, el uso de la memoria, los ciclos del reloj del procesador, etc.) mediante la cuantización de los valores de píxeles de los píxeles de escape de acuerdo con los aspectos de esta divulgación. En algunos ejemplos, la unidad de cuantización 106 puede implementar las técnicas descritas en el presente documento para cuantizar los valores de píxeles de escape sustituyendo operaciones de división con operaciones de desplazamiento (por ejemplo, operaciones de desplazamiento a la derecha). Más específicamente, la unidad de cuantización 106 puede determinar la operación específica de desplazamiento hacia la derecha basándose en el valor QP del píxel de escape correspondiente. Por ejemplo, la unidad de cuantización 106 puede derivar el valor de desplazamiento a la derecha resolviendo una función que incluya el valor QP como parámetro u operando.

35
 [0169] Por ejemplo, de acuerdo con las técnicas existentes, la unidad de codificación basada en paletas 122 puede formar una tabla que asigne el valor QP de cada píxel de escape a la cantidad de desplazamiento a la derecha para aplicar al valor del píxel. La unidad de codificación basada en paletas 122 puede formar la tabla para incluir 52 entradas. Por ejemplo, la tabla de asignación de 52 entradas puede proporcionar una cantidad de desplazamiento a la derecha correspondiente a cada posible valor de QP para un píxel de escape dado. A su vez, la unidad de cuantización 106 puede aplicar una operación de asignación para determinar la cantidad de desplazamiento a la derecha para cada píxel, basándose en la entrada del valor QP correspondiente en la tabla.

40
 [0170] En diversos ejemplos, la unidad de cuantización 106 puede implementar diversas técnicas de esta divulgación para cuantizar un píxel de escape mediante la determinación de la cantidad de desplazamiento hacia la derecha para un píxel de escape basándose en la aplicación de una función lineal al valor del píxel de escape para determinar la cantidad de desplazamiento a la derecha. Un ejemplo de una función lineal que la unidad de cuantización 106 puede aplicar para derivar la cantidad del desplazamiento a la derecha para cuantizar un píxel de escape es el siguiente:

$$\text{Right_shift} = a * ((QP + b) \gg c) + d,$$

50
 donde a, b, c y d son todos los parámetros de enteros. Además, el operador ">>" indica una operación de desplazamiento a la derecha. En un resultado específico de aplicar la ecuación anterior, la unidad de cuantización 106 puede determinar que la cantidad de desplazamiento a la derecha para un valor de píxel de escape es tres. La operación de desplazamiento a la derecha resultante se puede expresar como Desplazamiento a la derecha = (QP >> 3). Al resolver una función (por ejemplo, la función lineal descrita anteriormente), la unidad de cuantización 106 puede habilitar un dispositivo de decodificación (por ejemplo, el decodificador de vídeo 30 y/o diversos componentes del mismo) para descuantizar un píxel de escape, al tiempo que conserva los recursos de almacenamiento que de otro modo se utilizarían para almacenar la tabla de 52 entradas. De esta manera, la unidad de cuantización 106 puede implementar técnicas de esta divulgación para cuantizar un píxel de escape mientras mitiga los requisitos de almacenamiento con respecto al decodificador de vídeo 30.

60
 [0171] Las técnicas descritas en esta divulgación también pueden incluir técnicas para diversas combinaciones de uno o más modos de codificación basados en paletas de señalización, paletas de transmisión, paletas de predicción, paletas derivadas o mapas de codificación basados en paletas de transmisión y otros elementos sintácticos.

65
 [0172] La FIG. 3 es un diagrama de bloques que ilustra un decodificador de vídeo 30 a modo de ejemplo que está configurado para implementar las técnicas de esta divulgación. La FIG. 3 se proporciona para propósitos de explicación

y no se limita a las técnicas como las ampliamente ejemplificadas y descritas en esta divulgación. Para los propósitos de explicación, esta divulgación describe el decodificador de vídeo 30 en el contexto de la codificación HEVC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

5 **[0173]** En el ejemplo de la FIG. 3, el decodificador de vídeo 30 incluye una memoria de datos de vídeo 148, una unidad de decodificación por entropía 150, una unidad de procesamiento de predicción 152, una unidad de cuantización 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 decodificadas 162. La unidad de procesamiento de predicción 152 incluye una unidad de compensación de movimiento 164 y una unidad de procesamiento de intrapredicción 166. El decodificador de vídeo 30 también incluye una unidad de decodificación basada en paletas 165 configurada para realizar diversos aspectos de las técnicas de codificación basada en paletas descritas en esta divulgación. En otros ejemplos, el decodificador de vídeo 30 puede incluir más, menos o diferentes componentes funcionales.

15 **[0174]** La memoria de datos de vídeo 148 puede almacenar datos de vídeo, tales como un flujo de bits de vídeo codificado, para decodificarse por los componentes del decodificador de vídeo 30. Los datos de vídeo almacenados en la memoria de datos de vídeo 148 pueden obtenerse, por ejemplo, a partir del medio legible por ordenador 16, por ejemplo, desde una fuente de vídeo local, tal como una cámara, mediante comunicación de datos de vídeo por red cableada o inalámbrica, o accediendo a medios de almacenamiento físico de datos. La memoria de datos de vídeo 20 148 puede formar una memoria intermedia de imágenes codificadas (CPB) que almacena datos de vídeo codificados a partir de un flujo de bits de vídeo codificado. La memoria intermedia de imágenes decodificadas 162 puede ser una memoria de imágenes de referencia que almacene datos de vídeo de referencia para usar en la decodificación de datos de vídeo mediante el decodificador de vídeo 30, por ejemplo, en modos de intracodificación o de intercodificación. La memoria de datos de vídeo 148 y la memoria intermedia de imágenes decodificadas 162 pueden 25 estar formadas por cualquiera entre varios dispositivos de memoria, tales como memoria dinámica de acceso aleatorio (DRAM), incluyendo DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) u otros tipos de dispositivos de memoria. La memoria de datos de vídeo 148 y la memoria intermedia de imágenes decodificadas 162 pueden proporcionarse mediante el mismo dispositivo de memoria o dispositivos de memoria independientes. En diversos ejemplos, la memoria de datos de vídeo 148 puede estar en un chip con otros componentes del decodificador 30 de vídeo 30, o fuera de chip con respecto a esos componentes.

[0175] La memoria de datos de vídeo 148, es decir, una CPB, puede recibir y almacenar datos de vídeo codificados (por ejemplo, unidades NAL) de un flujo de bits. La unidad de decodificación por entropía 150 puede recibir datos de vídeo codificados (por ejemplo, unidades NAL) de la memoria de datos de vídeo 148 y puede analizar las unidades 35 NAL para decodificar elementos sintácticos. La unidad de decodificación por entropía 150 puede decodificar por entropía elementos sintácticos codificados por entropía en las unidades de NAL. La unidad de procesamiento de predicción 152, la unidad de cuantización 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 decodificados basándose en los elementos sintácticos obtenidos (por ejemplo, extraídos) del flujo de bits.

40 **[0176]** Las unidades NAL del flujo de bits pueden incluir unidades NAL de fragmentos codificados. Como parte de la decodificación del flujo de bits, la unidad de decodificación por entropía 150 puede extraer y decodificar por entropía elementos sintácticos de las unidades de NAL de fragmentos codificados. Cada uno de los fragmentos codificados puede incluir en encabezado de fragmento y datos de fragmento. El encabezado de fragmento puede contener 45 elementos sintácticos pertenecientes a un fragmento. Los elementos sintácticos en el encabezado de fragmento pueden incluir un elemento sintáctico que identifique un PPS asociado a una imagen que contenga el fragmento.

[0177] Además de decodificar los elementos sintácticos del flujo de bits, el decodificador de vídeo 30 puede realizar una operación de reconstrucción en una CU no dividida. Para realizar la operación de reconstrucción en una CU no dividida, el decodificador de vídeo 30 puede realizar una operación de reconstrucción en cada TU de la CU. Realizando la operación de reconstrucción para cada TU de la CU, el decodificador de vídeo 30 puede reconstruir bloques 50 residuales de la CU.

[0178] Como parte de la realización de una operación de reconstrucción en una TU de una CU, la unidad de cuantización inversa 154 puede cuantizar de forma inversa, es decir, descuantizar, los bloques de coeficientes asociados a la TU. La unidad de cuantización inversa 154 puede utilizar un valor de QP asociado a la CU de la TU para determinar un grado de cuantización y, asimismo, un grado de cuantización inversa, para aplicar por la unidad de cuantización inversa 154. Es decir, la razón de compresión, es decir, la razón entre el número de bits utilizados para representar la secuencia original y la comprimida, puede controlarse ajustando el valor del QP usado al cuantizar 60 los coeficientes de transformada. La razón de compresión también puede depender del procedimiento de codificación por entropía empleado.

[0179] Después de que la unidad de cuantización inversa 154 cuantiza inversamente un bloque de coeficientes, la unidad de procesamiento de transformada inversa 156 puede aplicar una o más transformadas inversas al bloque de coeficientes para generar un bloque de muestras residual asociado a la TU. Por ejemplo, la unidad de procesamiento de transformada inversa 156 puede aplicar una DCT inversa, una transformada entera inversa, una transformada de 65

Karhunen-Loeve (KLT) inversa, una transformada de rotación inversa, una transformada direccional inversa u otra transformada inversa al bloque de coeficientes.

5 **[0180]** Si se codifica una PU usando la intrapredicción, la unidad de procesamiento de intrapredicción 166 puede realizar la intrapredicción para generar bloques predictivos para la PU. La unidad de procesamiento de intrapredicción 166 puede usar una modalidad de intrapredicción para generar los bloques predictivos de luma, Cb y Cr para la PU basándose en los bloques predictivos de las PU espacialmente contiguas. La unidad de procesamiento de intrapredicción 166 puede determinar la modalidad de intrapredicción para la PU basándose en uno o más elementos sintácticos decodificados a partir del flujo de bits.

10 **[0181]** La unidad de procesamiento de predicción 152 puede construir una primera lista de imágenes de referencia (ListalmagenRef0) y una segunda lista de imágenes de referencia (ListalmagenRef1) basándose en los elementos sintácticos extraídos del flujo de bits. Además, si se codifica una PU usando la interpredicción, la unidad de decodificación por entropía 150 puede extraer información de movimiento para la PU. La unidad de compensación de movimiento 164 puede determinar, basándose en la información de movimiento de la PU, una o más regiones de referencia para la PU. La unidad de compensación de movimiento 164 puede generar, basándose en bloques de muestras en uno o más bloques de referencia para la PU, bloques predictivos (por ejemplo, bloques predictivos de luma, Cb y Cr) para la PU.

20 **[0182]** La unidad de reconstrucción 158 puede usar los bloques de transformada (por ejemplo, los bloques de transformada de luma, Cb y Cr) asociados con las TU de una CU y los bloques predictivos (por ejemplo, los bloques de Luma, Cb y Cr) de las PU de la UC, es decir, cualquiera de los datos de predicción o de los datos de interpredicción, según corresponda, para reconstruir los bloques de codificación (por ejemplo, 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 (por ejemplo, los bloques de transformada Luma, Cb y Cr) a las muestras correspondientes de los bloques predictivos (por ejemplo, los bloques de Luma, Cb y Cr predictivos) para reconstruir los bloques de codificación (por ejemplo, bloques de codificación luma, Cb y Cr) de la CU.

30 **[0183]** La unidad de filtro 160 puede realizar una operación de desbloqueo para reducir los artefactos de bloqueo asociados con los bloques de codificación (por ejemplo, los bloques de codificación de luma, Cb y Cr) de la CU. El decodificador de vídeo 30 puede almacenar los bloques de codificación (por ejemplo, los bloques de codificación de luma, Cb y Cr) de la CU en la memoria intermedia de imagen decodificada 162. La memoria intermedia de imágenes decodificadas 162 puede proporcionar imágenes de referencia para la posterior compensación de movimiento, la intrapredicción y la presentación en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1. Por ejemplo, el decodificador de vídeo 30 puede realizar, basándose en los bloques (por ejemplo, los bloques de luma, Cb y Cr) en la memoria intermedia de imágenes decodificadas 162, operaciones de intrapredicción o de interpredicción en PU de otras CU. De esta forma, el decodificador de vídeo 30 puede obtener, a partir del flujo de bits, los niveles de coeficientes de transformada de un bloque de coeficientes, cuantizar inversamente los niveles de coeficientes de transformada, aplicar una transformada a los niveles de coeficientes de transformada para generar un bloque de transformada, generar, basándose al menos en parte en el bloque de transformada, un bloque de codificación, y emitir el bloque de codificación para visualizar.

45 **[0184]** De acuerdo con diversos ejemplos de esta divulgación, el codificador de vídeo 30 puede configurarse para realizar una codificación basada en paletas. La unidad de codificación basada en paletas 165, por ejemplo, puede realizar una decodificación basada en paletas cuando se seleccione una modalidad de codificación basada en paletas, por ejemplo, para una CU o PU. Por ejemplo, la unidad de decodificación basada en paletas 165 puede configurarse para generar una paleta que tenga entradas que indiquen valores de píxeles. Además, en este ejemplo, la unidad de decodificación basada en paletas 165 puede recibir información que asocie al menos algunas posiciones de un bloque de datos de vídeo con entradas en la paleta. En este ejemplo, la unidad de decodificación basada en paletas 165 puede seleccionar valores de píxeles en la paleta basándose en la información. Adicionalmente, en este ejemplo, la unidad de decodificación basada en paletas 165 puede reconstruir los valores de píxeles del bloque basándose en los valores de píxeles seleccionados. Aunque se describen diversas funciones como realizadas por la unidad de decodificación basada en paletas 165, algunas de, o todas, estas funciones pueden realizarse por otras unidades de procesamiento, o por una combinación de diferentes unidades de procesamiento.

55 **[0185]** La unidad de decodificación basada en paletas 165 puede recibir información de modalidad de codificación por paleta, y realizar las operaciones anteriores cuando la información de modalidad de codificación de paleta indique que la modalidad de codificación por paleta se aplica al bloque. Cuando la información del modo de codificación por paleta indica que el modo de codificación por paleta no se aplica al bloque, o cuando la información de otro modo indica el uso de un modo diferente, la unidad de decodificación basada en paletas 165 decodifica el bloque de datos de vídeo usando un modo de codificación no basada en paletas, por ejemplo, dicho modo de codificación HEVC interpredictivo o intrapredictivo, cuando la información del modo de codificación por paleta indica que el modo de codificación de la paleta no se aplica al bloque. El bloque de datos de vídeo puede ser, por ejemplo, una CU o PU generada de acuerdo con un proceso de codificación de HEVC. Un codificador de vídeo 30 puede codificar algunos bloques con modos de predicción temporal interpredictiva o codificación espacial intrapredictiva y decodificar otros bloques con el modo de codificación basada en paletas. El modo de codificación basada en paletas puede comprender

uno de una pluralidad de diferentes modos de codificación basada en paletas, o puede haber un único modo de codificación basada en paletas.

5 **[0186]** De acuerdo con una o más de las técnicas de esta divulgación, el decodificador de vídeo 30, y específicamente la unidad de decodificación basada en paletas 165, puede realizar la decodificación de vídeo basada en paletas de bloques de vídeo codificados por paleta. Como se describió anteriormente, una paleta decodificada por el decodificador de vídeo 30 puede codificarse y señalizarse explícitamente por el codificador de vídeo 20, reconstruida por el decodificador de vídeo 30 con respecto a un bloque codificado por paleta recibido, predicha a partir de entradas de paleta anteriores, predicha a partir de valores de píxeles anteriores una combinación de los mismos.

10 **[0187]** Como se describió anteriormente, si un bloque de vídeo codificado por paleta cumple con un conjunto particular de condiciones, el codificador de vídeo 20 (y diversos componentes del mismo, tal como la unidad de codificación basada en paletas 122) puede omitir la codificación y la señalización de un mapa de valores de índice de paleta para los píxeles del bloque. En los ejemplos en los que el codificador de vídeo 20 omite la codificación y la señalización del mapa de valores de índice de paleta para un único bloque de color codificado por paleta, el decodificador de vídeo 30 (y sus componentes específicos, tal como la unidad de decodificación basada en paleta 165) puede aplicar diversas técnicas de esta divulgación para reconstruir el bloque de un único color. En algunos ejemplos, la unidad de decodificación basada en paletas 165 puede realizar operaciones recíprocas a las descritas anteriormente con respecto a la unidad de codificación basada en paletas 122 del codificador de vídeo 20, para determinar que el bloque codificado por paleta es un bloque de un único color. Por ejemplo, la unidad de decodificación basada en paletas 165 puede determinar que la paleta para el bloque actual tiene un tamaño de uno, lo que determina que el bloque cumple con la primera condición para clasificarse como un bloque de un único color. En diversos ejemplos, el decodificador de vídeo 30 puede recibir la paleta en un flujo de bits de vídeo codificado del codificador de vídeo 20, o puede derivar la paleta usando diversos otros datos incluidos en el flujo de bits de vídeo codificado.

25 **[0188]** Adicionalmente, el decodificador de vídeo 30 puede determinar que el bloque no incluye ningún píxel de escape, determinando de este modo que el bloque cumpla con la segunda condición para clasificarse como un bloque de un único color. Basándose en la determinación de que el tamaño de la paleta para el bloque es uno (cumpliendo por tanto con la primera condición), y que el bloque no incluye ningún píxel de escape (cumpliendo por tanto con la segunda condición), la unidad de decodificación basada en paletas 165 puede implementar técnicas de esta divulgación para determinar que el bloque actual es un bloque de un único color. A su vez, la unidad de decodificación basada en paletas 165 puede reconstruir el bloque de un único color mediante la reconstrucción de todos los píxeles del bloque de un único color de acuerdo con la información de color indicada en la entrada única de la paleta correspondiente. De esta manera, la unidad de decodificación basada en paletas 165 puede implementar técnicas de esta divulgación para reconstruir un bloque codificado por paleta con precisión, al tiempo que conserva los recursos informáticos y el ancho de banda que de otro modo se requeriría para reconstruir el bloque basándose en un mapa de valores de índice de paleta para todos los píxeles del bloque.

40 **[0189]** En algunos ejemplos, el decodificador de vídeo 30 puede recibir, en el flujo de bits de vídeo codificado, un indicador que indique si el codificador de vídeo 20 omitió la codificación y la señalización del mapa de valores de índice de paleta para uno o más píxeles de un bloque codificado de paleta, de acuerdo con técnicas de esta divulgación. En los casos en que el decodificador de vídeo 30 recibe un indicador que indica que el codificador de vídeo 20 omitió la codificación y la señalización del mapa de valores de índice de paleta para el bloque codificado por paleta, la unidad de decodificación basada en paletas 165 puede implementar técnicas de esta divulgación para determinar que el bloque actual está codificado por paleta y es un bloque de un único color. Más específicamente, si el indicador está habilitado (por ejemplo, establecido en un valor de uno), la unidad de decodificación basada en paletas 165 puede determinar que el bloque codificado por paleta es un bloque de un único color. A su vez, la unidad de decodificación basada en paletas 165 puede implementar técnicas de esta divulgación para reconstruir todos los píxeles del bloque de acuerdo con la información de color de la única entrada en la paleta para el bloque. Por tanto, la unidad de decodificación basada en paletas 165 puede implementar técnicas de esta divulgación para reconstruir con precisión el bloque codificado por paleta usando un indicador de un bit para todo el bloque, en lugar de usar valores de índice independientes (con una profundidad de bits variable) para diferentes píxeles o grupos de píxeles (por ejemplo, una línea), del bloque. De esta manera, la unidad de decodificación basada en paletas 165 puede conservar la utilización de recursos informáticos en el decodificador de vídeo 30 en la reconstrucción de bloques codificados por una paleta de un único color, y puede reducir el ancho de banda requerido por el decodificador de vídeo 30 para recibir los datos necesarios para reconstruir los bloques codificados de paleta de un único color, manteniendo la precisión y la calidad de imagen.

60 **[0190]** Como se describe, el codificador de vídeo 20 (y los componentes del mismo, tal como la unidad de codificación basada en paletas 122 y/o la unidad de cuantización 106) pueden implementar ciertas técnicas de esta divulgación para cuantizar los valores de píxeles de escape de un bloque codificado por paleta con una mayor eficiencia informática. El decodificador de vídeo 30 (y diversos componentes del mismo, tales como la unidad de decodificación basada en paletas 165 y/o la unidad de cuantización inversa 154) pueden realizar operaciones recíprocas a las descritas anteriormente con respecto al codificador de vídeo 20, para cuantizar los píxeles de escape de acuerdo con diversas técnicas de esta divulgación. Por ejemplo, la unidad de cuantización inversa 154 puede descuantizar todos los píxeles de escape de un único canal de color usando el mismo valor QP, basándose en la

información recibida en el flujo de bits de vídeo codificado del codificador de vídeo 20. Más específicamente, de acuerdo con los aspectos de esta divulgación, la unidad de cuantización inversa 154 puede descuantizar cualquier píxel de escape (o errores de predicción/valores residuales de los mismos) comunicados a través de un canal de color particular, usando un valor de QP que se determina basándose en el valor de QP para la descuantización de coeficiente de transformada tradicional para bloques comunicados a través del canal de color actual. En algunos ejemplos, la unidad de cuantización inversa 154 puede implementar las técnicas de esta divulgación para descuantizar los píxeles de escape comunicados a través de diferentes canales de color usando diferentes valores de QP, basándose en el valor de QP usado para la codificación del coeficiente de transformada tradicional, que es diferente entre los diferentes canales.

[0191] De esta manera, el decodificador de vídeo 30 puede implementar las técnicas descritas en el presente documento para definir y aplicar un único valor de QP (para descuantizar) todos los píxeles de escape comunicados a través de un canal de color particular. Por tanto, el decodificador de vídeo 30 puede aplicar aspectos de esta divulgación para definir un valor QP para píxeles de escape detectados a través de la codificación basada en paletas, donde las técnicas de codificación existentes basada en paletas no definieron un valor QP para píxeles de escape.

[0192] En algunos ejemplos, los componentes del decodificador de vídeo 30, tales como la unidad de cuantización inversa 154, pueden implementar técnicas de esta divulgación para realizar operaciones recíprocas de las descritas anteriormente con respecto al codificador de vídeo 20 (y/o componentes del mismo, tales como la unidad de cuantización 106), para cuantizar un valor de píxel de escape cuantizado. Por ejemplo, la unidad de cuantización inversa 154 puede implementar técnicas de esta divulgación para calcular una cantidad de desplazamiento (por ejemplo, para una operación de desplazamiento a la izquierda correspondiente) basándose en un valor de QP para descuantizar el valor de píxel de escape cuantizado correspondiente. De esta manera, la unidad de cuantización inversa 154 también puede aplicar aspectos de esta divulgación para conservar los recursos informáticos, tal como la utilización del almacenamiento, al aprovechar una función en lugar de almacenar una tabla de asignación de 52 entradas.

[0193] La FIG. 4 es un diagrama de flujo que ilustra un proceso 180 de ejemplo mediante el cual un dispositivo de decodificación de vídeo puede implementar técnicas de esta divulgación para omitir la decodificación de valores de índice para píxeles de un bloque codificado por paleta, basándose en un conjunto particular de condiciones. Aunque el proceso 180 puede realizarse mediante una variedad de dispositivos de acuerdo con aspectos de esta divulgación, el proceso 180 se describe en el presente documento con respecto al decodificador de vídeo 30 de las FIGS. 1 y 3, para los propósitos de facilitar la descripción. El proceso 180 puede comenzar cuando el decodificador de vídeo 30 determine una cantidad de entradas incluidas en una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a decodificar (182). Adicionalmente, el decodificador de vídeo 30 puede determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no está asociado con ninguna de las entradas de la paleta (184). Por ejemplo, si la información de color de un píxel del bloque no se asigna a ninguna entrada de paleta, el decodificador de vídeo 30 puede identificar dicho píxel como un píxel de escape. En diversos ejemplos, el decodificador de vídeo 30 puede identificar el píxel de escape usando un indicador señalado por el codificador de vídeo 20 o un valor de índice (por ejemplo, "otro índice" descrito anteriormente) señalado por el codificador de vídeo 20.

[0194] A su vez, el decodificador de vídeo 30 puede, en respuesta a la determinación de que el número de entradas incluidas en la paleta es igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, omitir la decodificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo (186). Como ejemplo, el decodificador de vídeo 30 puede recibir, como parte de un flujo de bits de vídeo codificado, datos de vídeo codificados, por ejemplo, elementos sintácticos y/o indicadores, asociados con el bloque de datos de vídeo, en el que los datos de vídeo codificados asociados con el bloque no incluya valores de índice asociados con la paleta para los valores de píxeles del bloque. Además, el decodificador de vídeo 30 puede determinar los valores de píxeles del bloque de datos de vídeo para que sean iguales a la entrada incluida en la paleta (188). Por ejemplo, el decodificador de vídeo 30 puede reconstruir el bloque asignando a todos los píxeles del bloque la información de color indicada por la entrada única de la paleta.

[0195] En un ejemplo, el decodificador de vídeo 30 puede recibir además, como parte de un flujo de bits de vídeo codificado, un indicador que indique si los valores de índice están codificados para los datos de vídeo de bloque. En un ejemplo, para determinar el número de entradas incluidas en la paleta, el decodificador de vídeo 30 puede recibir, como parte de un flujo de bits de vídeo codificado, un indicador que indique si el número de entradas en la paleta es igual a uno. En un ejemplo, para determinar si el bloque de datos de vídeo incluye al menos un píxel de escape, el decodificador de vídeo 30 puede recibir, como parte de un flujo de bits de vídeo codificado, un indicador que indique si el bloque de datos de vídeo incluye al menos un píxel de escape. En un ejemplo, el decodificador de vídeo 30 puede recibir, como parte de un flujo de bits de vídeo codificado, uno o más elementos sintácticos asociados con la paleta. En este ejemplo, el decodificador de vídeo 30 puede decodificar uno o más elementos sintácticos asociados con la paleta.

[0196] La FIG. 5 es un diagrama de flujo que ilustra un proceso 200 de ejemplo mediante el cual un dispositivo de codificación de vídeo puede implementar técnicas de esta divulgación para derivar la codificación de valores de índice

para píxeles de un bloque codificado por paleta, basándose en un conjunto particular de condiciones. Si bien el proceso 200 puede realizarse por una variedad de dispositivos de acuerdo con aspectos de esta divulgación, el proceso 200 se describe en el presente documento con respecto al codificador de vídeo 20 de las FIGS. 1 y 2, con el propósito de facilitar la descripción. El proceso 200 puede comenzar cuando el codificador de vídeo 20 determine una cantidad de entradas incluidas en una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar (202). Adicionalmente, el codificador de vídeo 20 puede determinar si el bloque de datos de vídeo incluye al menos un píxel de escape que no está asociado con ninguna de las entradas en la paleta (204).

[0197] A su vez, el codificador de vídeo 20, en respuesta a la determinación de que el número de entradas incluidas en la paleta es igual a uno y que el bloque de datos de vídeo no incluye al menos un píxel de escape, determina que los valores de píxeles del bloque son iguales a la entrada de paleta, y omite la codificación de los valores de índice asociados con la paleta para los valores de píxeles del bloque de datos de vídeo (206). Por ejemplo, el dispositivo de codificación de vídeo 20 puede codificar datos, por ejemplo, elementos sintácticos y/o indicadores, para el bloque sin codificar los valores de índice que relacionan los valores de píxeles del bloque con las entradas en la paleta correspondiente al bloque. Adicionalmente, el dispositivo de codificación de vídeo 20 puede codificar uno o más elementos sintácticos asociados con el bloque de datos de vídeo (208).

[0198] En un ejemplo, para codificar uno o más elementos sintácticos, el codificador de vídeo 20 puede codificar, como parte de un flujo de bits de vídeo codificado, un indicador que indica si los valores de índice están codificados para los valores de píxeles del bloque de datos de vídeo. En un ejemplo, para codificar uno o más elementos sintácticos, el codificador de vídeo 20 puede codificar, como parte de un flujo de bits de vídeo codificado, un indicador que indica si el tamaño de la paleta es igual a uno. En un ejemplo, para codificar uno o más elementos sintácticos, el codificador de vídeo 20 puede codificar, como parte de un flujo de bits de vídeo codificado, un indicador que indique si el bloque de datos de vídeo incluye al menos un píxel de escape. En un ejemplo, el codificador de vídeo 20 puede codificar uno o más elementos sintácticos asociados con la paleta. En este ejemplo, el codificador de vídeo 20 puede señalar, como parte de un flujo de bits de vídeo codificado, uno o más elementos sintácticos asociados con la paleta.

[0199] La FIG. 6 es un diagrama de flujo que ilustra un proceso 220 de ejemplo mediante el cual un dispositivo de decodificación de vídeo puede implementar técnicas de esta divulgación para descuantizar uno o más píxeles de escape de un bloque de datos de vídeo codificado por paleta. Aunque el proceso 220 puede realizarse mediante una variedad de dispositivos de acuerdo con aspectos de esta divulgación, el proceso 220 se describe en el presente documento con respecto al decodificador de vídeo 30 de las FIGS. 1 y 3, para los propósitos de facilitar la descripción. El proceso 220 puede comenzar cuando el decodificador de vídeo 30 determine una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a decodificar (222). Adicionalmente, el decodificador de vídeo 30 puede identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de una o más entradas en la paleta (224).

[0200] A su vez, el decodificador de vídeo 30 puede identificar un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de QP usado para la codificación del coeficiente de transformada en la codificación no basada en paletas (226). Por ejemplo, el decodificador de vídeo 30 puede determinar que el único valor de QP es igual al valor de QP usado para la decodificación de coeficiente tradicional de un canal de color asociado con el bloque. Además, el decodificador de vídeo 30 puede descuantizar cada uno de los uno o más píxeles de escape usando el único valor de QP (228) identificado. A su vez, el decodificador de vídeo 30 puede determinar los valores de píxeles para el bloque de datos de vídeo basándose en los píxeles de escape descuantizados y los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que están asociados con una o más entradas en la paleta (230).

[0201] En un ejemplo, dos entradas cualquiera de la paleta varían al menos en un límite de error de paleta. En un ejemplo, el límite de error de paleta es directamente proporcional a un valor QP de paleta asociado con el bloque. En un ejemplo, para identificar el uno o más píxeles de escape, el decodificador de vídeo 30 puede recibir, en un flujo de bits de vídeo codificado, un indicador de un bit asociado con cada uno de los uno o más píxeles de escape, y determinar, basándose en el valor de cada indicador de un bit recibida, que cada uno de los uno o más píxeles de escape no está asociado con ninguna de las entradas de la paleta. En un ejemplo, para determinar que cada uno de los uno o más píxeles de escape no está asociado con ninguna de las entradas de la paleta, el decodificador de vídeo 30 puede determinar que cada uno de los uno o más píxeles de escape no está asociado con ninguna de las entradas de la paleta basándose en el valor de cada indicador de un bit recibido y sin decodificar un valor de otro índice predefinido asociado con píxeles de escape.

[0202] La FIG. 7 es un diagrama de flujo que ilustra un proceso 240 de ejemplo mediante el cual un dispositivo de codificación de vídeo puede implementar técnicas de esta divulgación para cuantizar uno o más píxeles de escape de un bloque de datos de vídeo codificado por paleta. Si bien el proceso 240 puede realizarse por una variedad de dispositivos de acuerdo con aspectos de esta divulgación, el proceso 240 se describe en el presente documento con respecto al codificador de vídeo 20 de las FIGS. 1 y 2, con el propósito de facilitar la descripción. El proceso 200 puede comenzar cuando el codificador de vídeo 20 determine una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar (242). Además, el codificador de vídeo 20 puede identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de una o más entradas

en la paleta (244).

[0203] A su vez, el codificador de vídeo 20 puede identificar un único valor de parámetro de cuantización (QP) para todos los uno o más píxeles de escape del bloque (246). Por ejemplo, el codificador de vídeo 20 puede determinar que el único valor de QP es igual a un valor de QP usado para la codificación de coeficiente tradicional de un canal de color asociado con el bloque. Adicionalmente, el codificador de vídeo 20 puede cuantizar cada uno de los uno o más píxeles de escape usando el único valor de QP identificado (228).

[0204] En un ejemplo, para identificar el único valor de QP, el codificador de vídeo 20 puede determinar que el único valor de QP es igual a un valor de QP usado para la codificación de coeficiente tradicional de un canal de color asociado con el bloque. En un ejemplo, el codificador de vídeo 20 puede determinar que un límite de error de paleta de la paleta es directamente proporcional al valor de QP asociado con el bloque, donde cualquiera de las dos entradas de la paleta varía al menos en un límite de error de paleta. En un ejemplo, para determinar el límite de error de paleta, el codificador de vídeo 20 puede identificar el límite de error de paleta usando una tabla que asigne el límite de error de paleta al valor de QP asociado con el bloque. En un ejemplo, el codificador de vídeo 20 puede codificar un indicador de un bit asociado con cada uno de los uno o más píxeles de escape sin codificar otro valor de índice predefinido asociado con los píxeles de escape, en el que un valor de cada indicador de un bit indica que un respectivo de uno o más píxeles de escape no está asociado con ninguna de las entradas de la paleta. En un ejemplo, para cuantizar cada uno de los uno o más píxeles de escape usando el único valor de QP identificado, el codificador de vídeo 20 puede resolver una función que se base en el único valor de QP identificado. En uno de estos ejemplos, para resolver la función, el codificador de vídeo 20 puede realizar una operación de desplazamiento hacia la derecha que se base en el único valor de QP identificado.

[0205] En algunos ejemplos, las técnicas para la codificación basada en paletas de datos de vídeo pueden usarse con una o más técnicas de codificación diferentes, tales como técnicas para la codificación inter o intrapredictiva. Por ejemplo, como se describe en mayor detalle a continuación, un codificador o decodificador, o un codificador-decodificador combinado (códec), se puede configurar para realizar una codificación interpredictiva e intrapredictiva, así como una codificación basada en paletas.

[0206] Debe reconocerse que, dependiendo del ejemplo, ciertos actos o sucesos de cualquiera de las técnicas descritas en el presente documento pueden realizarse en una secuencia distinta, pueden añadirse, 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 hilos, 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 realizarse por una combinación de unidades o módulos asociados a un codificador de vídeo.

[0207] Ciertos aspectos de esta divulgación se han descrito con respecto a la norma de HEVC en desarrollo, con fines de ilustración. Sin embargo, las técnicas descritas en esta divulgación pueden ser útiles para otros procesos de codificación de vídeo, que incluyen otros procesos de codificación de vídeo, estándar o patentados, aún no desarrollados.

[0208] 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), de los cuales ambos pueden denominarse en general codificadores de vídeo. Del mismo modo, la codificación de vídeo puede referirse a la codificación de vídeo o a la decodificación de vídeo, según corresponda.

[0209] Si bien se describen anteriormente combinaciones particulares de diversos aspectos de las técnicas, estas combinaciones se proporcionan meramente para ilustrar ejemplos de las técnicas descritas en esta divulgación. Por consiguiente, las técnicas de esta divulgación no deberían limitarse a estas combinaciones de ejemplo y pueden abarcar cualquier combinación concebible de los diversos aspectos de las técnicas descritas en esta divulgación.

[0210] En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o en cualquier combinación de estos. Si se implementan en software, las funciones pueden almacenarse o transmitirse, como una o más instrucciones o código, en o a través de un medio legible por ordenador y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que correspondan a un medio tangible tal como unos medios de almacenamiento de datos o unos medios de comunicación que incluyan cualquier medio que facilite la transferencia de un programa informático de un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder en general a (1) unos medios de almacenamiento tangibles legibles por ordenador que sean no transitorios, o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que uno o más ordenadores o uno o más procesadores puedan acceder para recuperar instrucciones, código y/o estructuras de datos para implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

5 **[0211]** A modo de ejemplo, y no de limitación, dichos medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar un 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 página web, un servidor u otra fuente remota mediante un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital (DSL) o unas 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 están incluidas 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 ni otros medios transitorios, sino que, en su lugar, se dirigen a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye un disco compacto (CD), un disco láser, un disco óptico, un disco versátil digital (DVD), un disco flexible y un disco Blu-ray, de los cuales los discos habitualmente reproduce datos magnéticamente, mientras que el resto de discos reproducen datos ópticamente con láseres. Las combinaciones de lo anterior también deben incluirse dentro del alcance de los medios legibles por ordenador.

20 **[0212]** Las instrucciones pueden ejecutarse por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de uso general, circuitos integrados específicos de la aplicación (ASIC), matriz de puertas programables por campo (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 codificar y decodificar, o incorporados en un códec combinado. Además, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.

30 **[0213]** Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un teléfono inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). En la presente divulgación se describen diversos componentes, módulos o unidades para poner de manifiesto aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no se requiere necesariamente su realización por 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, incluidos uno o más procesadores como se ha descrito anteriormente, junto con software y/o firmware adecuados.

35 **[0214]** Se han descrito diversos ejemplos. La invención se define en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un procedimiento (220) de decodificación de datos de vídeo, el procedimiento que comprende:
- 5 determinar (222) una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a decodificar, en el que cualquiera de las dos entradas de la paleta varía al menos un límite de error de la paleta de modo que, si dos entradas posibles están dentro del límite de error de la paleta, una de las entradas se use para representar ambas entradas, y
- 10 en el que el límite de error de la paleta es directamente proporcional a un valor de parámetro de cuantización de paleta asociado con el bloque; que identifica (224), en el bloque de datos de vídeo, uno o más píxeles de escape que no están asociados con ninguna de una o más entradas en la paleta;
- 15 identificar (226) un único valor de parámetro de cuantización para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de parámetro de cuantización usado para la codificación del coeficiente de transformada en la codificación no basada en paletas;
- 20 descuantizar (228) todos los píxeles de escape del canal de color dado usando el valor del parámetro de cuantización único identificado; y
- determinar (230) los valores de píxeles del bloque de datos de vídeo basándose en los píxeles de escape descuantizados y los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con las una o más entradas en la paleta.
- 25 2. Un procedimiento (240) de codificación de datos de vídeo, el procedimiento que comprende:
- determinar (242) una paleta usada para representar valores de píxeles de un bloque de datos de vídeo que se vaya a codificar;
- 30 determinar que un límite de error de paleta de la paleta es directamente proporcional a un valor de parámetro de cuantización de paleta asociado con el bloque, en el que el límite de error de paleta es tal que, si dos entradas posibles están dentro del límite de error de paleta, se use una de las entradas para representar ambas entradas,
- 35 en el que cualquiera de las dos entradas de la paleta varían al menos un límite de error de paleta;
- identificar (244), en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de las una o más entradas en la paleta; identificar (246) un único valor de parámetro de cuantización para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de parámetro de cuantización usado para la codificación del coeficiente de transformada en la codificación no basada en paletas;
- 40 cuantizar (248) todos los píxeles de escape del canal de color dado usando el valor del parámetro de cuantización único identificado; y
- 45 codificar (250) los valores de píxeles del bloque de datos de vídeo incluidos los píxeles de escape cuantizados y los valores de índice para uno o más píxeles en el bloque de datos de vídeo que estén asociados con la una o más entradas en la paleta.
- 50 3. El procedimiento según la reivindicación 2, en el que identificar el valor de parámetro de cuantización único comprende:
- determinar que el valor de parámetro de cuantización único es igual al valor de un parámetro de cuantización usado para la codificación de coeficiente tradicional de un canal de color asociado con el bloque.
- 55 4. El procedimiento según la reivindicación 3, en el que determinar el límite de error de paleta comprende:
- identificar el límite de error de la paleta usando una tabla que asigna el límite de error de la paleta al valor del parámetro de cuantización asociado con el bloque.
- 60 5. El procedimiento según la reivindicación 2, en el que cuantizar cada uno de los uno o más píxeles de escape que usan el valor del parámetro de cuantización único identificado comprende resolver una función que se basa en el valor del parámetro de cuantización único identificado.
- 65 6. El procedimiento según la reivindicación 5, en el que resolver la función comprende realizar una operación de desplazamiento a la derecha que se basa en el valor del parámetro de cuantización único identificado.

7. Un dispositivo para codificar datos de vídeo, el dispositivo que comprende:
- una memoria configurada para almacenar datos de vídeo; y
 - 5 uno o más procesadores en comunicación con la memoria y configurados para:
 - determinar una paleta usada para representar los valores de píxeles de un bloque de datos de vídeo que se vaya a codificar,
 - 10 en el que cualquiera de las dos entradas de la paleta varía en al menos un límite de error de paleta tal que, si dos entradas posibles están dentro del límite de error de la paleta, una de las entradas se use para representar ambas entradas, y
 - 15 en el que el límite de error de paleta es directamente proporcional a un valor de parámetro de cuantización de paleta asociado con el bloque;
 - identificar, en el bloque de datos de vídeo, uno o más píxeles de escape que no estén asociados con ninguna de las una o más entradas en la paleta;
 - 20 identificar un único valor de parámetro de cuantización para todos los uno o más píxeles de escape del bloque para un canal de color dado basándose en un valor de parámetro de cuantización usado para la codificación del coeficiente de transformada en la codificación no basada en paletas;
 - 25 aplicar el valor del parámetro de cuantización individual identificado a todos los píxeles de escape del canal de color dado; y
 - determinar los valores de píxeles del bloque de datos de vídeo basándose en los píxeles de escape y los valores de índice recibidos para uno o más píxeles en el bloque de datos de vídeo que estén asociados con las una o más entradas.
 - 30 **8.** El dispositivo según la reivindicación 7, en el que el dispositivo comprende un dispositivo de decodificación de vídeo, y en el que, para aplicar el valor del parámetro de cuantización único identificado a cada uno de los uno o más píxeles de escape, el uno o más procesadores están configurados para descuantizar cada uno de los uno o más píxeles de escape usando el valor de parámetro de cuantización único identificado.
 - 35 **9.** El dispositivo según la reivindicación 7, en el que para aplicar el valor del parámetro de cuantización único identificado a cada uno de los uno o más píxeles de escape, el uno o más procesadores están configurados para resolver una función que comprenda el desempeño de una operación de desplazamiento a la derecha basada en el valor de parámetro de cuantización único identificado, en el que la función.
 - 40 **10.** El dispositivo según la reivindicación 7, en el que el dispositivo comprende un dispositivo para codificar datos de vídeo, y en el que los uno o más procesadores están configurados además para:
 - 45 identificar el límite de error de la paleta usando una tabla que asigne el límite de error de la paleta al valor del parámetro de cuantización asociado con el bloque.
 - 50 **11.** Un medio de almacenamiento legible por ordenador codificado con instrucciones que, cuando se ejecutan, causan que uno o más procesadores de un dispositivo informático realicen el procedimiento de las reivindicaciones 1-6.

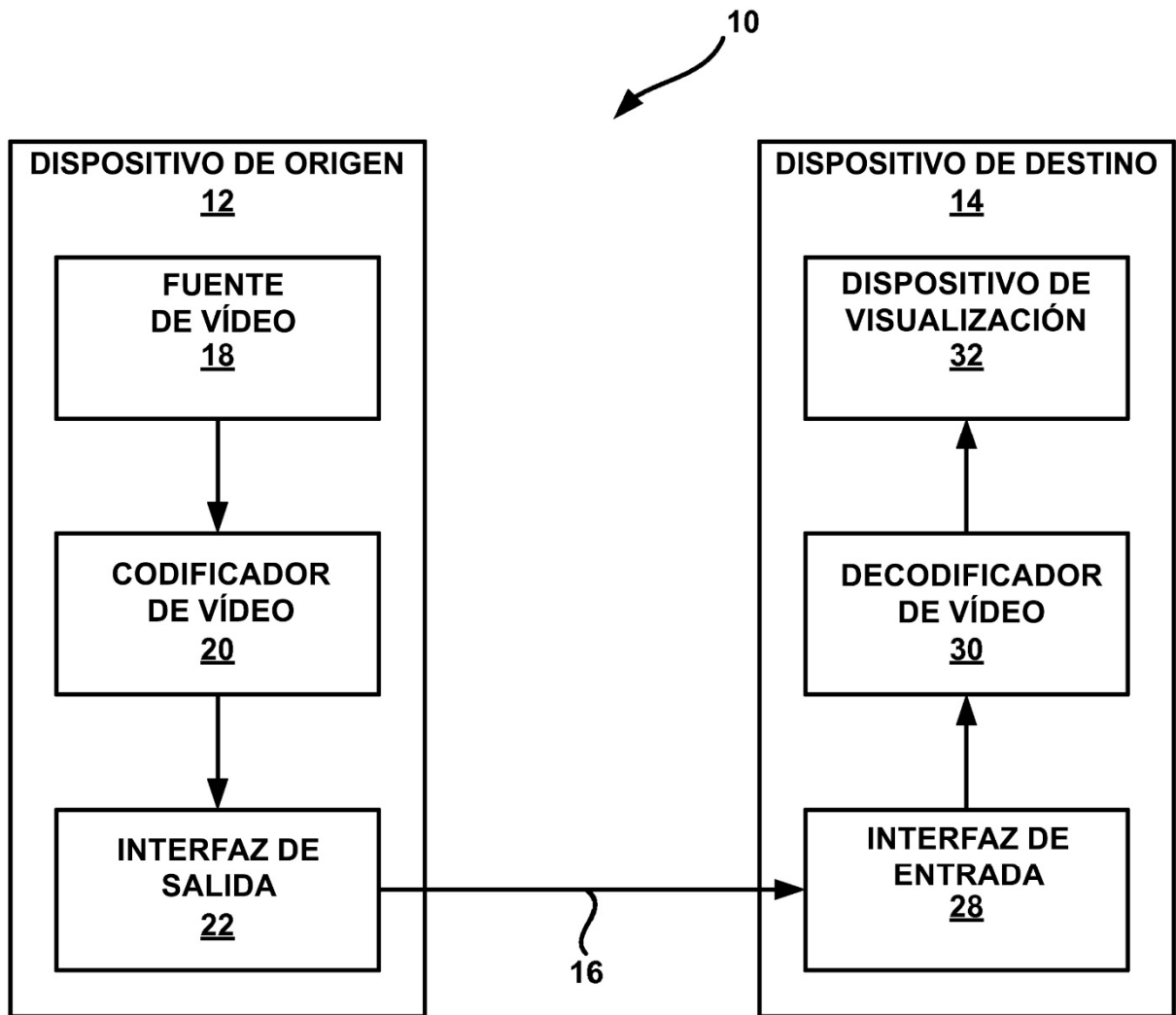


FIG. 1

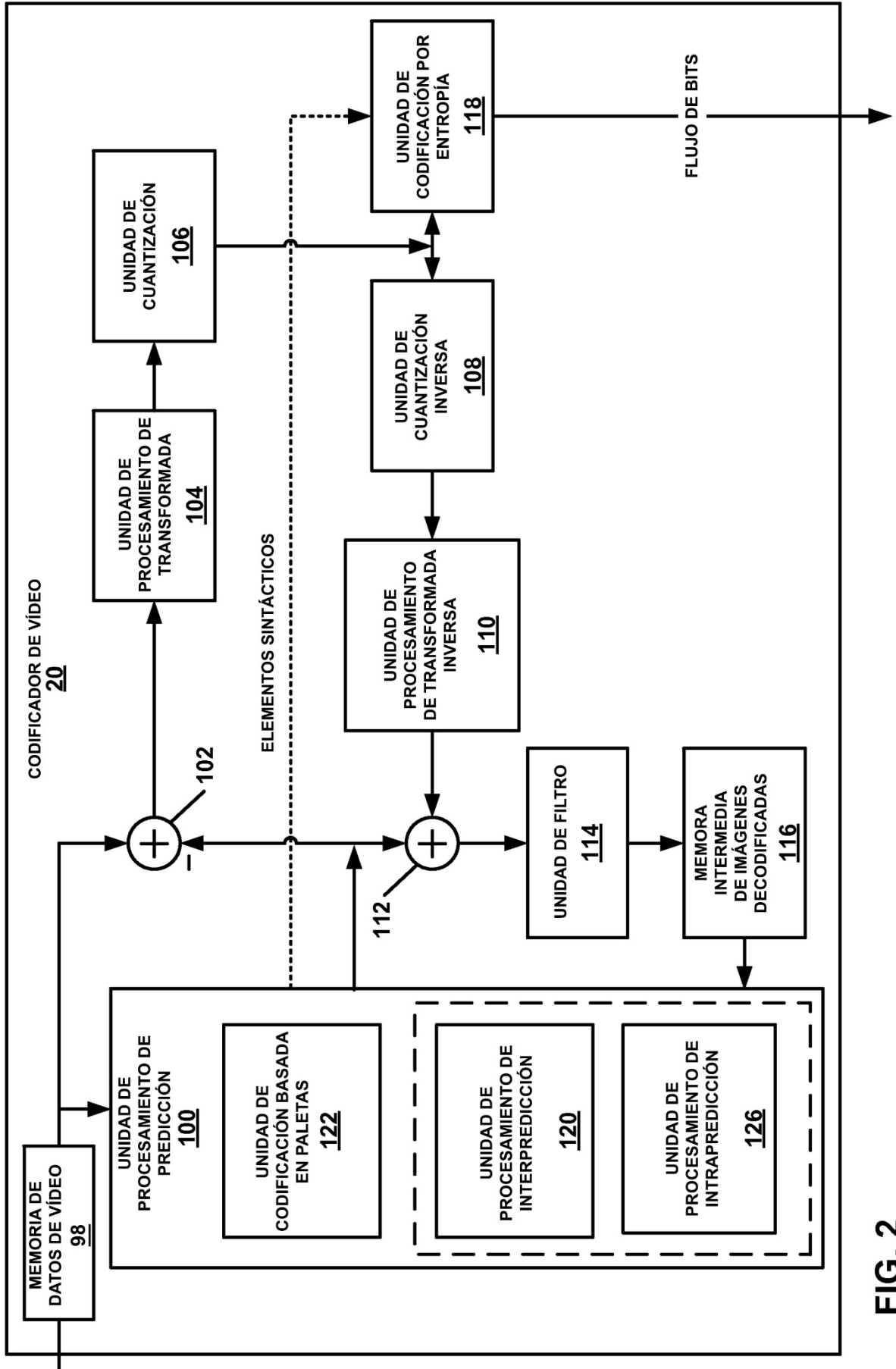


FIG. 2

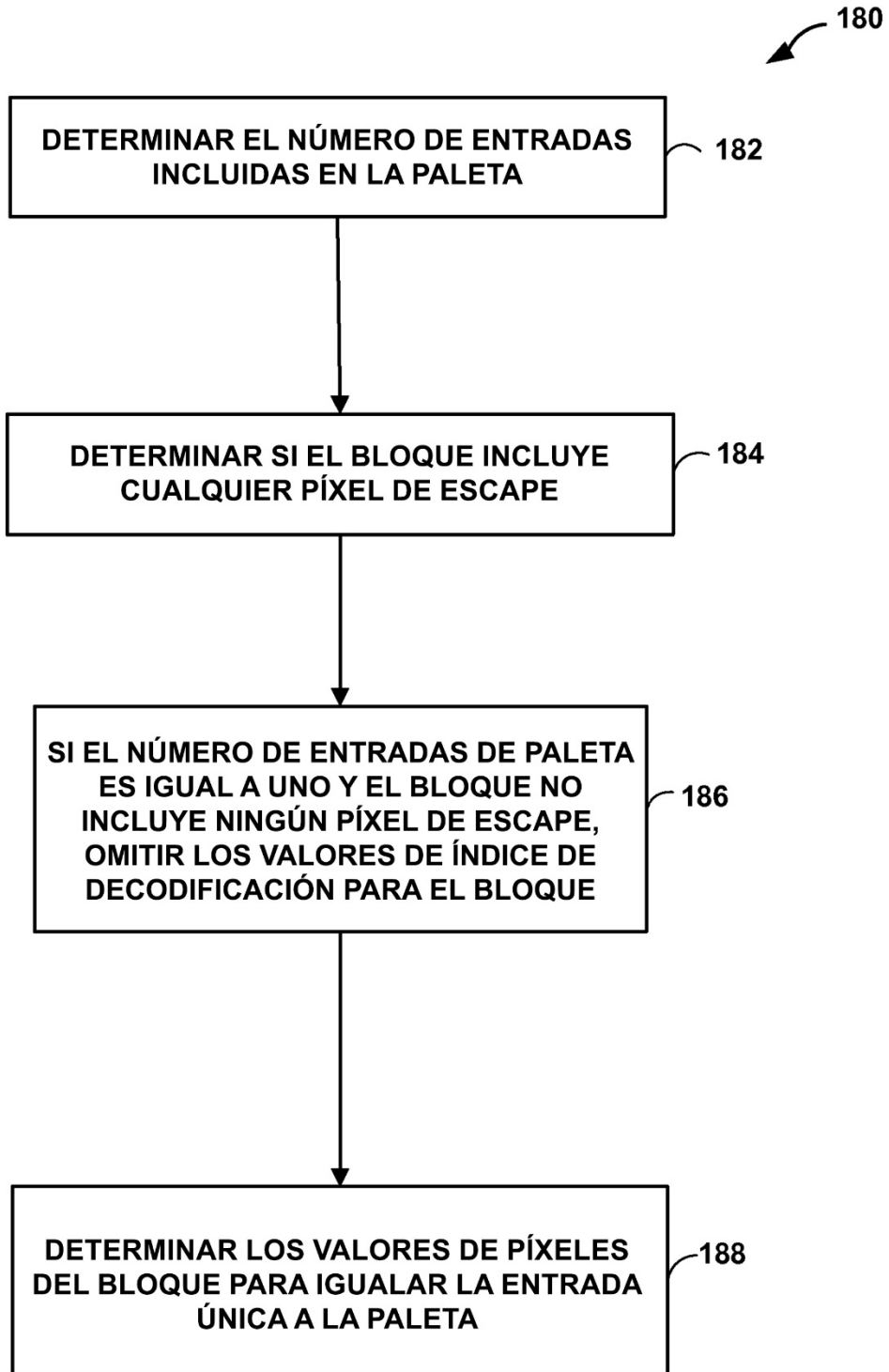


FIG. 4

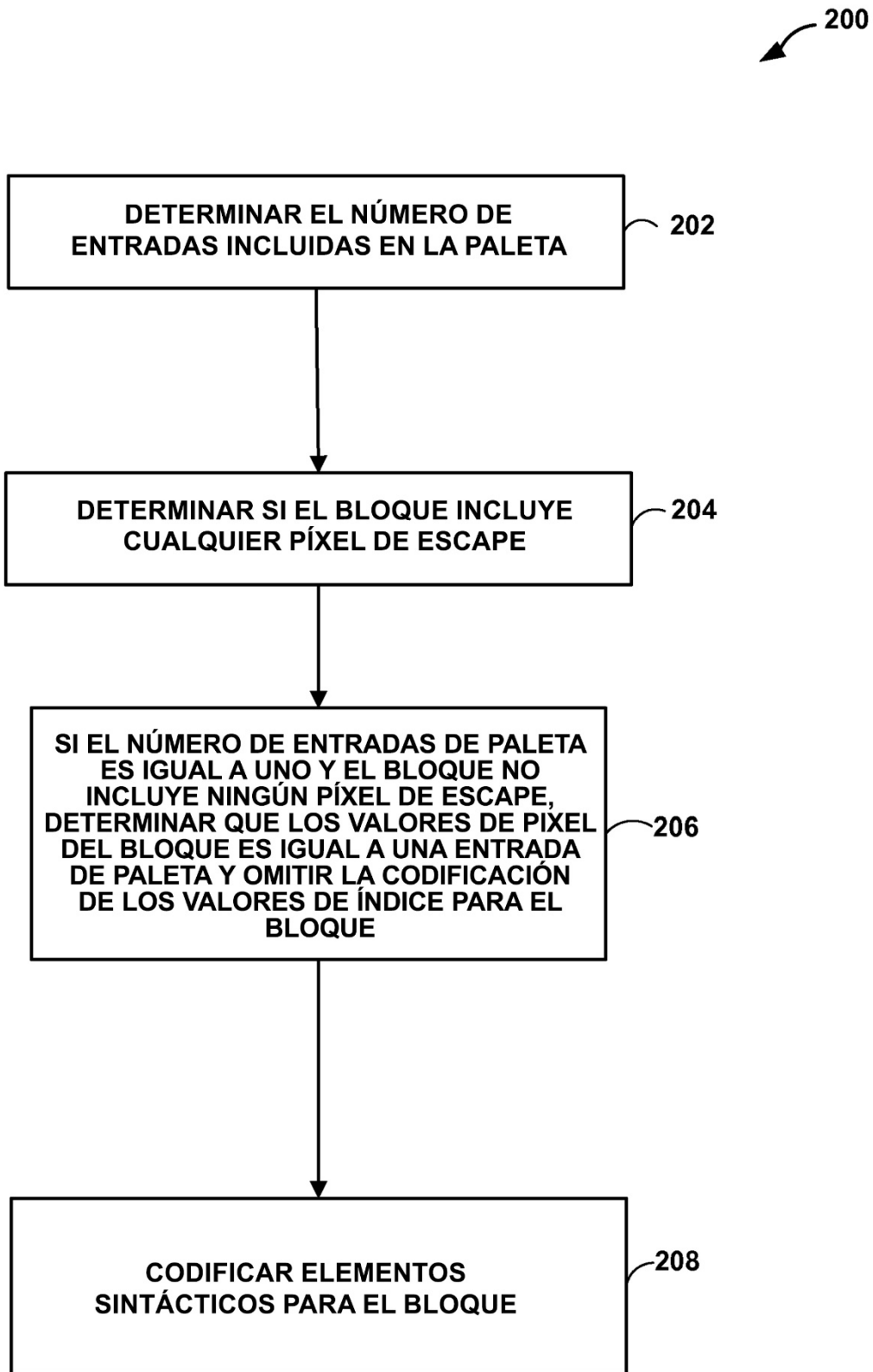


FIG. 5

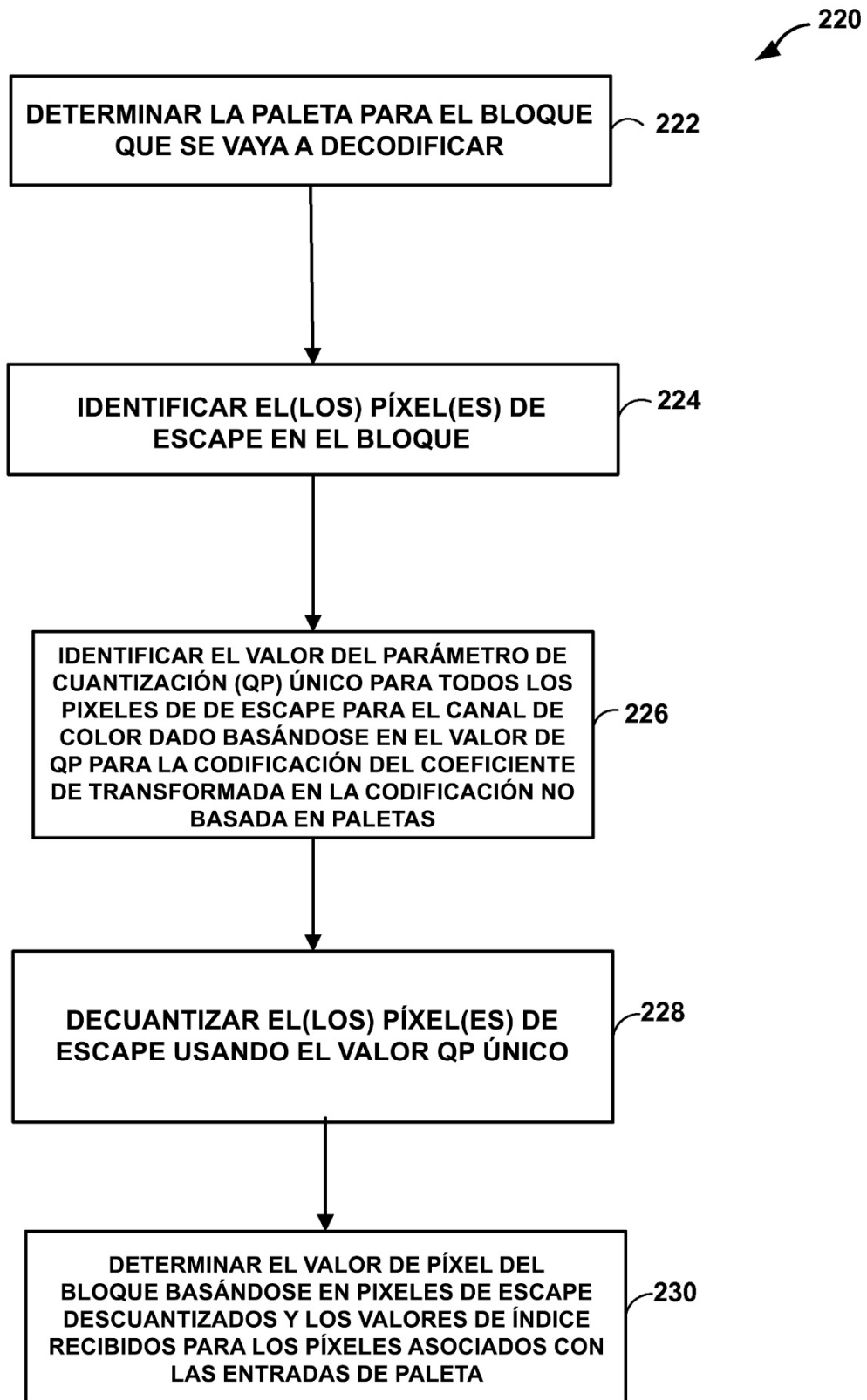


FIG. 6

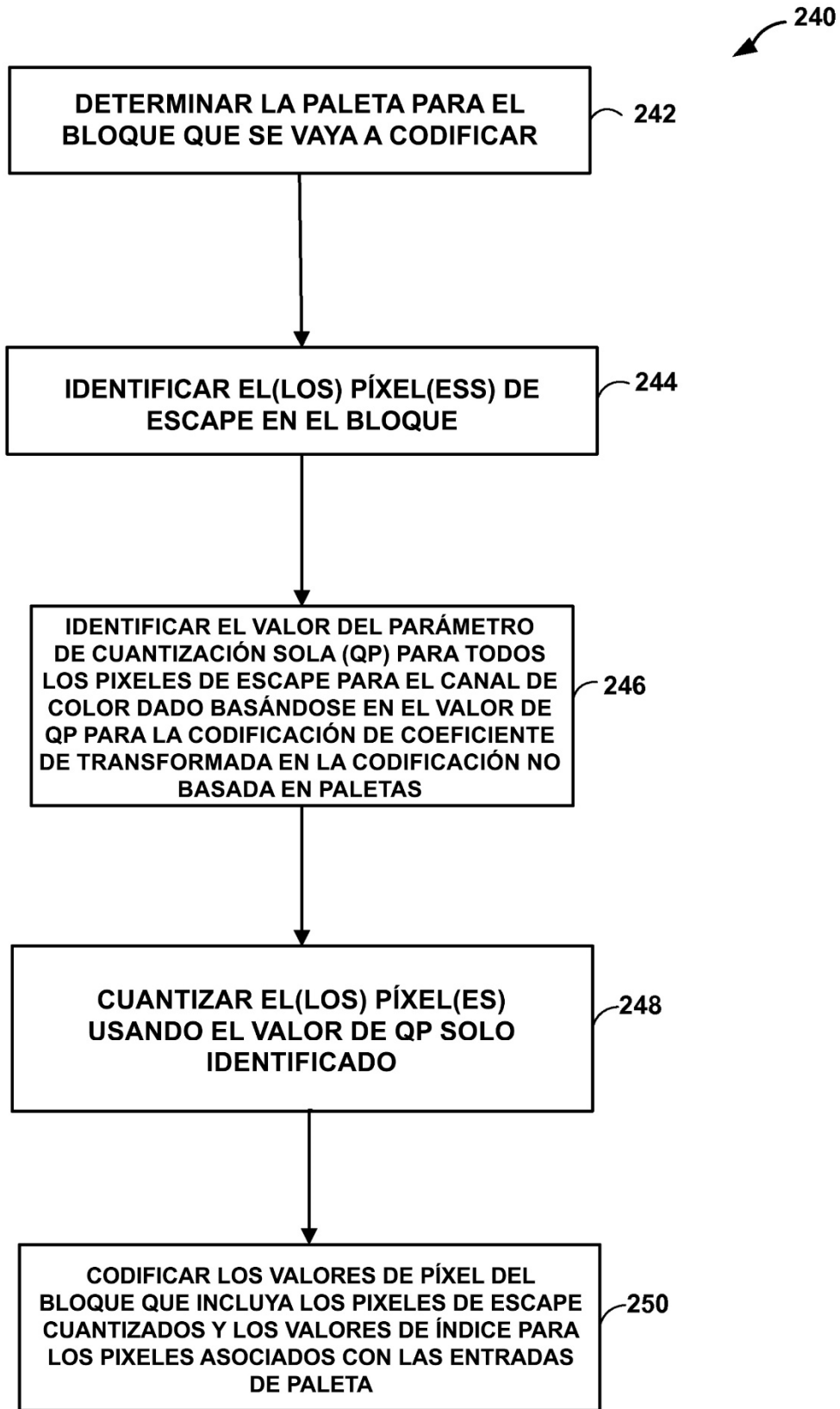


FIG. 7