

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 702 213**

51 Int. Cl.:

H04N 19/124 (2014.01)

H04N 19/176 (2014.01)

H04N 19/115 (2014.01)

H04N 19/14 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **14.04.2015 PCT/US2015/025744**

87 Fecha y número de publicación internacional: **22.10.2015 WO15160803**

96 Fecha de presentación y número de la solicitud europea: **14.04.2015 E 15778783 (9)**

97 Fecha y número de publicación de la concesión europea: **12.09.2018 EP 3132603**

54 Título: **Sistema y procedimiento para la detección de planicidad para la compresión de flujo de pantalla (DSC)**

30 Prioridad:

15.04.2014 US 201461979980 P

07.08.2014 US 201462034693 P

17.12.2014 US 201462093381 P

13.04.2015 US 201514685479

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
27.02.2019

73 Titular/es:

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

72 Inventor/es:

**THIRUMALAI, VIJAYARAGHAVAN;
JACOBSON, NATAN HAIM y
JOSHI, RAJAN LAXMAN**

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 702 213 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

Sistema y procedimiento para la detección de planicidad para la compresión de flujo de pantalla (DSC)

5 **CAMPO TÉCNICO**

[0001] Esta divulgación se refiere al campo de la codificación y compresión de vídeo y, en particular, a la compresión de vídeo para su transmisión por enlaces de visualización, como la compresión de flujo de pantalla (DSC).

10 **ANTECEDENTES**

Descripción de la técnica relacionada

15 [0002] Las capacidades de vídeo digital se pueden incorporar a una amplia gama de pantallas, que incluyen televisores digitales, asistentes digitales personales (PDA), ordenadores portátiles, monitores de escritorio, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o por radio satelital, dispositivos de videoconferencia y similares. Los enlaces de pantalla se utilizan para conectar pantallas a dispositivos de origen adecuados. Los requisitos de ancho de banda de los enlaces de pantalla son proporcionales a la resolución de las pantallas y, por lo tanto, las pantallas de alta resolución requieren enlaces de pantalla de gran ancho de banda. Algunos enlaces de pantalla no tienen el ancho de banda para prestar soporte a pantallas de alta resolución. La compresión de vídeo se puede usar para reducir los requisitos de ancho de banda, de modo que los enlaces de visualización de menor ancho de banda se puedan usar para proporcionar vídeo digital a pantallas de alta resolución.

25 [0003] Otros han intentado utilizar la compresión de imágenes en los datos de píxeles. Sin embargo, tales esquemas a veces no son visualmente libres de pérdidas o pueden ser difíciles y costosos de implementar en dispositivos de visualización convencionales.

30 [0004] La Asociación de Normas de Electrónica de Vídeo (VESA) ha desarrollado la compresión de flujo de pantalla (DSC) como norma para la compresión de vídeo de enlace de pantalla. La técnica de compresión de vídeo de enlace de pantalla, como DSC, debería proporcionar, entre otras cosas, una calidad de imagen que no tenga pérdidas visuales (es decir, lo suficientemente buena como para que los usuarios no puedan distinguir que la compresión está activa). La técnica de compresión de vídeo de enlace de pantalla también debería proporcionar un esquema que sea fácil y económico de implementar en tiempo real con hardware convencional.

35 [0005] El documento US 5.440.344 se divulga adaptar un nivel de cuantización para cada bloque en función de los valores de actividad por bloque a fin de obtener una tasa de bits constante.

40 **RESUMEN**

[0006] Cada uno de los sistemas, procedimientos y dispositivos de esta divulgación tiene varios aspectos innovadores, ninguno de los cuales es el único responsable de los atributos deseables divulgados en el presente documento.

45 [0007] Aspectos de la presente invención se exponen en las reivindicaciones independientes.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0008]

50 La FIG. 1A es un diagrama de bloques que ilustra un ejemplo de sistema de codificación y decodificación de vídeo que puede utilizar técnicas según los aspectos descritos en esta divulgación.

55 La FIG. 1B es un diagrama de bloques que ilustra otro ejemplo de sistema de codificación y decodificación de vídeo que puede llevar a cabo técnicas según los aspectos descritos en esta divulgación.

La FIG. 2A es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede implementar técnicas según aspectos descritos en esta divulgación.

60 La FIG. 2B es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo que puede implementar técnicas según aspectos descritos en esta divulgación.

65 La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de una serie de píxeles utilizados en los procedimientos de detección de planicidad de acuerdo con aspectos descritos en esta divulgación.

La FIG. 4 es un diagrama de bloques que ilustra un ejemplo de un detector de planicidad que puede realizar técnicas de acuerdo con aspectos descritos en esta divulgación.

5 La FIG. 5 es un diagrama de flujo que ilustra un procedimiento para la detección de planicidad de acuerdo con aspectos descritos en esta divulgación.

DESCRIPCIÓN DETALLADA

10 **[0009]** En general, esta divulgación se refiere a procedimientos de detección de planicidad en datos de vídeo en el contexto de técnicas de compresión de vídeo tales como la compresión de flujo de pantalla (DSC). Más específicamente, la presente divulgación se refiere a sistemas y procedimientos para identificar un bloque de transición dentro de los datos de vídeo dentro de los cuales se produce una transición desde una región compleja de los datos de vídeo a una región plana de los datos de vídeo.

15 **[0010]** Si bien ciertas formas de realización se describen en el presente documento en el contexto de la norma DSC, alguien medianamente experto en la técnica apreciará que los sistemas y procedimientos descritos en el presente documento pueden ser aplicables a cualquier norma de codificación de vídeo adecuada. Por ejemplo, formas de realización divulgadas en el presente documento pueden ser aplicables a una o más de las siguientes normas: Unión Internacional de las Telecomunicaciones (UIT) - Sector de Normalización de las Telecomunicaciones (UIT-T) H.261, Visual del Grupo de Expertos en Imágenes en Movimiento-1 (MPEG-1) de la Organización Internacional de Normalización / Comisión Electrotécnica Internacional (ISO/CEI), UIT-T H.262 o Visual del MPEG-2 de la ISO/CEI, UIT-T H.263, Visual del MPEG-4 de la ISO/CEI, UIT-T H.264 (también conocida como AVC del MPEG-4 de la ISO/CEI) y Codificación de Vídeo de Alta Eficacia (HEVC) y extensiones de dichas normas. Además, las técnicas descritas en esta divulgación pueden formar parte de normas desarrolladas en el futuro. En otras palabras, las técnicas descritas en esta divulgación pueden ser aplicables a normas de codificación de vídeo desarrolladas previamente, normas de codificación de vídeo actualmente en desarrollo y normas inminentes de codificación de vídeo.

30 **[0011]** La norma DSC incluye la detección de transiciones en datos de vídeo desde regiones complejas a regiones planas y desde regiones planas a regiones complejas. Dichas transiciones en la planicidad de los datos de vídeo pueden ser utilizadas por la DSC para determinar los parámetros de codificación usados en la codificación de los datos de vídeo. Estos parámetros de codificación se determinan, en parte, de manera que el vídeo comprimido cumpla con los requisitos de ancho de banda de la norma DSC y minimice la distorsión en el vídeo.

35 **[0012]** Las técnicas de compresión de vídeo, tales como DSC, pueden incluir detección de planicidad para detectar la transición desde una región compleja de los datos de vídeo a una región plana de los datos de vídeo. En algunos procedimientos de detección de planicidad, cada grupo de los datos de vídeo o cada bloque de datos de vídeo se señala mediante un parámetro asociado en el flujo de bits que define si el grupo es "complejo" o "plano" y, además, cuando se señala como plano, el tipo de planicidad se indica en cuanto a si el grupo es algo plano o muy plano. En consecuencia, la información señalizada puede aumentar los datos incluidos en los datos de vídeo comprimidos. Además, estas técnicas de detección de planicidad pueden basarse en los valores máximo y mínimo de los píxeles considerados (por ejemplo, el píxel anterior, el grupo actual de píxeles y/o el siguiente grupo de píxeles). Como tal, estas técnicas de detección de planicidad pueden no ajustarse bien a escala hasta tamaños más grandes de grupo o de bloque, ya que los valores máximo y mínimo pueden no capturar información con respecto a, por ejemplo, las variaciones locales dentro del grupo y, como tal, pueden no representar adecuadamente la planicidad del grupo.

45 **[0013]** En consecuencia, al menos un aspecto de esta divulgación se refiere a un procedimiento de detección de planicidad que incluye un cálculo de complejidad de cada uno entre un bloque anterior, un bloque actual y un bloque siguiente a los datos de vídeo. La planicidad del bloque actual se puede determinar en función de las complejidades de los bloques anterior, actual y siguiente. Por ejemplo, la planicidad puede ser una medida de la textura y/o la frecuencia espacial del bloque actual. Por consiguiente, puede ser posible ajustar a escala el tamaño del bloque mientras se captura adecuadamente la información relacionada con las variaciones locales en los bloques. Además, se puede identificar un bloque de transición donde ocurre la transición desde una región compleja a una región plana.

Normas de Codificación de Vídeo

55 **[0014]** Una imagen digital, tal como una imagen de vídeo, una imagen de televisión, una imagen fija o una imagen generada por una grabadora de vídeo o un ordenador, puede consistir en píxeles o muestras dispuestas en líneas horizontales y verticales. La cantidad de píxeles en una sola imagen suele ser de decenas de miles. Cada píxel contiene habitualmente información de luminancia y crominancia. Sin compresión, la mera cantidad de información que se transmitirá desde un codificador de imágenes a un decodificador de imágenes haría nada práctica la transmisión de imágenes en tiempo real. Para reducir la cantidad de información a transmitir, se han desarrollado una serie de procedimientos de compresión diferentes, tales como las normas JPEG, MPEG y H.263.

65 **[0015]** Las normas de codificación de vídeo incluyen UIT-T H.261, Visual del MPEG-1 de la ISO/IEC, UIT-T H.262 o Visual del MPEG-2 de la ISO/IEC, UIT-T H.263, Visual del MPEG-4 de la ISO/IEC, UIT -T H.264 (también conocida como AVC del MPEG-4 de la ISO/IEC) y HEVC, incluidas las extensiones de dichas normas.

[0016] Además, VESA ha desarrollado una norma de codificación de vídeo, a saber, DSC. La norma DSC es una norma de compresión de vídeo que puede comprimir vídeo para su transmisión por enlaces de pantalla. A medida que aumenta la resolución de las pantallas, el ancho de banda de los datos de vídeo requeridos para controlar las pantallas aumenta de manera correspondiente. Es posible que algunos enlaces de pantalla no tengan el ancho de banda para transmitir todos los datos de vídeo a la pantalla para tales resoluciones. En consecuencia, la norma DSC especifica una norma de compresión para la compresión interoperable, sin pérdida visual, sobre los enlaces de pantalla. La norma DSC 1.0 fue publicada en 2014 por VESA. Actualmente se están considerando avances adicionales en la DSC.

[0017] La norma DSC es diferente a otras normas de codificación de vídeo, tales como H.264 y HEVC. La DSC incluye compresión intratramas, pero no incluye compresión entre tramas, lo que significa que la norma DSC no puede usar información temporal para codificar los datos de vídeo. Por el contrario, otras normas de codificación de vídeo pueden emplear compresión entre tramas en sus técnicas de codificación de vídeo.

Sistema de Codificación de Vídeo

[0018] En lo sucesivo se describen más completamente diversos aspectos de los sistemas, aparatos y procedimientos novedosos, con referencia a los dibujos adjuntos. Sin embargo, esta divulgación se puede integrar de muchas formas diferentes y no se debería interpretar que está limitada a cualquier estructura o función específica presentada a lo largo de esta divulgación. En cambio, estos aspectos se proporcionan para que esta divulgación sea exhaustiva y completa, y transmita por completo el alcance de la divulgación a los expertos en la técnica. Basándose en las enseñanzas en el presente documento, un experto en la materia debería apreciar que el alcance de la divulgación pretende abarcar cualquier aspecto de los sistemas, aparatos y procedimientos novedosos divulgados en el presente documento, ya sean implementados de forma independiente de, o en combinación con, cualquier otro aspecto de la presente divulgación. Por ejemplo, un aparato se puede implementar o un procedimiento se puede llevar a la práctica usando cualquier número de los aspectos expuestos en el presente documento. Además, el alcance de la presente divulgación pretende abarcar dicho aparato o procedimiento que se lleva a la práctica usando otra estructura, funcionalidad o estructura y funcionalidad, además, o aparte, de los diversos aspectos de la presente divulgación expuestos en el presente documento. Debería entenderse que cualquier aspecto divulgado en el presente documento puede integrarse mediante uno o más elementos de una reivindicación.

[0019] Aunque en el presente documento se describen aspectos particulares, muchas variantes y permutaciones de estos aspectos están dentro del alcance de la divulgación. Aunque se mencionan algunos beneficios y ventajas de los aspectos preferidos, el alcance de la divulgación no está concebido para limitarse a beneficios, usos u objetivos particulares. En cambio, los aspectos de la divulgación pretenden ser ampliamente aplicables a diferentes tecnologías inalámbricas, configuraciones de sistema, redes y protocolos de transmisión, algunos de los cuales se ilustran a modo de ejemplo en las figuras y en la siguiente descripción de los aspectos preferidos. La descripción detallada y los dibujos son meramente ilustrativos de la divulgación, en lugar de ser limitantes, estando definido el alcance de la divulgación por las reivindicaciones adjuntas y equivalentes de las mismas.

[0020] Los dibujos adjuntos ilustran ejemplos. Los elementos indicados mediante números de referencia en los dibujos adjuntos corresponden a elementos indicados mediante números de referencia similares en la siguiente descripción. En esta divulgación, los elementos que tienen nombres que comienzan con palabras ordinales (por ejemplo, "primero", "segundo", "tercero", etc.) no necesariamente implican que los elementos tienen un orden particular. Más bien, dichas palabras ordinales se usan simplemente para referirse a diferentes elementos de un mismo tipo o un tipo similar.

[0021] La FIG. 1A es un diagrama de bloques que ilustra un ejemplo de sistema de codificación de vídeo 10 que puede utilizar técnicas de acuerdo con aspectos descritos en esta divulgación. Como se usa en este documento, el término "codificador de vídeo" o "codificador" 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 y a la decodificación de vídeo. Además de los codificadores de vídeo y los decodificadores de vídeo, los aspectos descritos en la presente solicitud pueden extenderse a otros dispositivos relacionados, tales como los transcodificadores (por ejemplo, dispositivos que pueden decodificar un flujo de bits y recodificar otro flujo de bits) y cajas intermedias (por ejemplo, dispositivos que pueden modificar, transformar y/o manipular de otro modo un flujo de bits).

[0022] Como se muestra en la FIG. 1A, el sistema de codificación de vídeo 10 incluye un dispositivo de origen 12 que genera datos de vídeo codificados para ser decodificados posteriormente por un dispositivo de destino 14. En el ejemplo de la FIG. 1A, el dispositivo de origen 12 y el dispositivo de destino 14 constituyen dispositivos independientes. Sin embargo, se observa que el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar en, o ser parte de, el mismo dispositivo, como se muestra en el ejemplo de la FIG. 1B.

[0023] Con referencia una vez más a la FIG. 1A, el dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender, respectivamente, cualquiera entre una amplia gama de dispositivos, incluidos los ordenadores de escritorio, los ordenadores plegables (por ejemplo, los ordenadores portátiles), los ordenadores de tableta, los

5 decodificadores, los auriculares telefónicos tales como los llamados "teléfonos inteligentes", los llamados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión de flujo de vídeo o similares. En algunas formas de realización, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

10 [0024] El dispositivo de destino 14 puede recibir, a través del enlace 16, los datos de vídeo codificados a decodificar. El enlace 16 puede comprender cualquier tipo de medio o dispositivo capaz de desplazar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En el ejemplo de la **FIG. 1A**, el enlace 16 puede comprender un medio de comunicación para permitir que el dispositivo de origen 12 transmita datos de vídeo codificados al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados pueden modularse de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitirse al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación, inalámbrica o cableada, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física. El medio de comunicación puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área amplia o una red global tal como Internet. El medio de comunicación puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 14.

20 [0025] En el ejemplo de la **FIG. 1A**, el dispositivo de origen 12 incluye un origen de vídeo 18, un codificador de vídeo 20 y la interfaz de salida 22. En algunos casos, la interfaz de salida 22 puede incluir un modulador/demodulador (módem) y/o un transmisor. En el dispositivo de origen 12, el origen de vídeo 18 puede incluir un origen tal como un dispositivo de captura de vídeo, por ejemplo, una videocámara, un archivo de vídeo que contiene vídeo previamente capturado, una interfaz de alimentación de vídeo para recibir vídeo desde un proveedor de contenido de vídeo y/o un sistema de gráficos de ordenador para generar datos de gráficos de ordenador como el vídeo de origen, o una combinación de tales orígenes. Como un ejemplo, si el origen de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los denominados "teléfonos de cámara" o "videoteléfonos", tal como se ilustra en el ejemplo de la **FIG. 1B**. Sin embargo, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y pueden aplicarse a aplicaciones inalámbricas y/o cableadas.

30 [0026] El vídeo capturado, precapturado o generado por ordenador puede ser codificado por el codificador de vídeo 20. Los datos de vídeo codificados pueden ser transmitidos al dispositivo de destino 14 mediante la interfaz de salida 22 del dispositivo de origen 12. Los datos de vídeo codificados pueden almacenarse también (o de forma alternativa) en el dispositivo de almacenamiento 31 para un acceso posterior por parte del dispositivo de destino 14 u otros dispositivos, para su decodificación y/o reproducción. El codificador de vídeo 20 ilustrado en las **FIG. 1A** y **1B** puede comprender el codificador de vídeo 20 ilustrado en la **FIG. 2A** o cualquier otro codificador de vídeo descrito en este documento.

40 [0027] En el ejemplo de la **FIG. 1A**, el dispositivo de destino 14 incluye la interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. En algunos casos, la interfaz de entrada 28 puede incluir un receptor y/o un módem. La interfaz de entrada 28 del dispositivo de destino 14 puede recibir los datos de vídeo codificados por el enlace 16 y/o desde el dispositivo de almacenamiento 31. Los datos de vídeo codificados, comunicados por el enlace 16, o proporcionados en el dispositivo de almacenamiento 31, pueden incluir una diversidad de elementos sintácticos generados por el codificador de vídeo 20, para su uso por un decodificador de vídeo, tal como el decodificador de vídeo 30, en la decodificación de los datos de vídeo. Dichos elementos sintácticos pueden incluirse con los datos de vídeo codificados, transmitidos en un medio de comunicación, almacenados en un medio de almacenamiento o almacenados en un servidor de ficheros.

50 [0028] El dispositivo de visualización 32 puede estar integrado con, o ser externo a, el dispositivo de destino 14. En algunos ejemplos, el dispositivo de destino 14 puede incluir un dispositivo de visualización integrado y también estar configurado para interconectarse con un dispositivo de visualización externo. En otros ejemplos, el dispositivo de destino 14 puede ser un dispositivo de visualización. En general, el dispositivo de visualización 32 muestra los datos de vídeo decodificados a un usuario, y puede comprender cualquiera entre 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.

60 [0029] En aspectos relacionados, la **FIG. 1B** muestra un ejemplo de sistema de codificación de vídeo 10' en el que el dispositivo de origen 12 y el dispositivo de destino 14 están en, o forman parte de, un dispositivo 11. El dispositivo 11 puede ser un equipo telefónico de mano, tal como un teléfono "inteligente" o similar. El dispositivo 11 puede incluir un dispositivo controlador/procesador 13 (optativamente presente) en comunicación operativa con el dispositivo de origen 12 y el dispositivo de destino 14. El sistema de codificación de vídeo 10' de la **FIG. 1B**, y sus componentes, son similares de otro modo al sistema de codificación de vídeo 10 de la **FIG. 1A** y sus componentes.

65 [0030] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como DSC. De forma alternativa, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden funcionar de acuerdo con otras normas privadas o industriales, tales como la norma UIT-T H.264, denominada de forma alternativa MPEG-4, Parte 10, y AVC, HEVC o ampliaciones de dichas normas. Sin embargo, las técnicas

de esta divulgación no están limitadas a ninguna norma de codificación particular. Otros ejemplos de normas de compresión de vídeo incluyen MPEG-2 e UIT-T H.263.

[0031] Aunque no se muestra en los ejemplos de las **FIG. 1A** y **1B**, el codificador de vídeo 20 y el decodificador de vídeo 30 pueden estar integrados, cada uno, con un codificador y decodificador de audio, y pueden incluir unidades MUX-DEMUX apropiadas, u otro hardware y software, para gestionar la codificación tanto de audio como de vídeo en un flujo de datos común o flujos de datos por separado. Si procede, en algunos ejemplos, las unidades MUX-DEMUX pueden ser conformes al protocolo de multiplexador UIT H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

[0032] El codificador de vídeo 20 y el decodificador de vídeo 30 pueden implementarse, cada uno, como cualquiera entre una variedad de circuitos codificadores adecuados, tales como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de los mismos. Cuando las técnicas se implementan parcialmente en software, un dispositivo puede almacenar instrucciones para el software en un medio adecuado no transitorio legible por ordenador, y ejecutar las instrucciones en hardware mediante uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden estar incluidos en uno o más codificadores o decodificadores, cualquiera de los cuales puede estar integrado como parte de un codificador/decodificador combinado en un dispositivo respectivo.

Proceso de Codificación de Vídeo

[0033] Como se ha mencionado brevemente en lo que antecede, el codificador de vídeo 20 codifica datos de vídeo. Los datos de vídeo pueden comprender una o más imágenes. Cada una de las imágenes es una imagen fija que forma parte de un vídeo. En algunos casos, una imagen puede denominarse una "trama" de vídeo. Cuando el codificador de vídeo 20 codifica los datos de vídeo, el codificador de vídeo 20 puede generar un flujo de bits. El flujo de bits puede incluir una secuencia de bits que forman una representación codificada de los datos de vídeo. El flujo de bits puede incluir imágenes codificadas y datos asociados. Una imagen codificada es una representación codificada de una imagen.

[0034] Para generar el flujo de bits, el codificador de vídeo 20 puede llevar a cabo operaciones de codificación en cada imagen en los datos de vídeo. Cuando el codificador de vídeo 20 realiza operaciones de codificación en las imágenes, el codificador de vídeo 20 puede generar una serie de imágenes codificadas y datos asociados. Los datos asociados pueden incluir un conjunto de parámetros de codificación, tales como un parámetro de cuantización (QP). Para generar una imagen codificada, el codificador de vídeo 20 puede dividir una imagen en bloques de vídeo de igual tamaño. Un bloque de vídeo puede ser una formación bidimensional de muestras. Los parámetros de codificación pueden definir una opción de codificación (por ejemplo, un modo de codificación) para cada bloque de los datos de vídeo. La opción de codificación se puede seleccionar para lograr un rendimiento deseado de velocidad-distorsión.

[0035] En algunos ejemplos, el codificador de vídeo 20 puede dividir una imagen en una pluralidad de fragmentos. Cada uno de los fragmentos puede incluir una región espacialmente distinta en una imagen (por ejemplo, una trama) que puede decodificarse independientemente sin información del resto de las regiones en la imagen o trama. Cada imagen o trama de vídeo se puede codificar en un solo fragmento o se puede codificar en varios fragmentos. En la DSC, los bits de destino asignados para codificar cada fragmento pueden ser esencialmente constantes. Como parte de la realización de una operación de codificación en una imagen, el codificador de vídeo 20 puede realizar operaciones de codificación en cada fragmento de la imagen. Cuando el codificador de vídeo 20 realiza una operación de codificación en un fragmento, el codificador de vídeo 20 puede generar datos codificados asociados con el fragmento. Los datos codificados asociados con el fragmento pueden denominarse un "fragmento codificado".

Codificador de Vídeo DSC

[0036] La **FIG. 2A** es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo 20 que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El codificador de vídeo 20 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del codificador de vídeo 20. En algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar algunas de, o todas, las técnicas descritas en esta divulgación.

[0037] Con fines de explicación, esta divulgación describe el codificador de vídeo 20 en el contexto de la codificación según la norma DSC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

[0038] En el ejemplo de la **FIG. 2A**, el codificador de vídeo 20 incluye una pluralidad de componentes funcionales. Los componentes funcionales del codificador de vídeo 20 incluyen un convertidor de espacio cromático 105, un almacén temporal 110, un detector de planicidad 115, un controlador de velocidad 120, un componente predictor, cuantizador y reconstructor 125, un almacén temporal de línea 130, un historial cromático indizado 135, un codificador

por entropía 140, un multiplexor de subflujo 145 y un almacén temporal de velocidad 150. En otros ejemplos, el codificador de vídeo 20 puede incluir más, menos o diferentes componentes funcionales.

[0039] El convertidor de espacio cromático 105 puede convertir un espacio cromático de entrada en el espacio cromático utilizado en la implementación de codificación. Por ejemplo, en una forma de realización a modo de ejemplo, el espacio cromático de los datos de vídeo de entrada está en el espacio cromático de rojo, verde y azul (RGB) y la codificación se implementa en el espacio cromático de la luminancia Y, el verde de crominancia Cg y el naranja de crominancia Co (YCgCo). La conversión del espacio cromático se puede realizar mediante uno o varios procedimientos que incluyen desplazamientos y adiciones a los datos de vídeo. Se observa que los datos de entrada de vídeo en otros espacios cromáticos pueden procesarse y también se pueden realizar conversiones a otros espacios cromáticos.

[0040] En aspectos relacionados, el codificador de vídeo 20 puede incluir el almacén temporal 110, el almacén temporal de línea 130 y/o el almacén temporal de velocidad 150. Por ejemplo, el almacén temporal 110 puede contener los datos de vídeo convertidos del espacio cromático antes del uso de los datos de vídeo convertidos del espacio cromático por otras partes del codificador de vídeo 20. En otro ejemplo, los datos de vídeo pueden almacenarse en el espacio cromático RGB y la conversión de color-espacio puede realizarse según sea necesario, ya que los datos convertidos del espacio cromático pueden requerir más bits.

[0041] El almacén temporal de velocidad 150 puede funcionar como parte del mecanismo de control de velocidad en el codificador de vídeo 20, que se describirá con mayor detalle a continuación en relación con el controlador de velocidad 120. Los bits empleados en la codificación de cada bloque pueden variar muy considerablemente en función de la naturaleza del bloque. El almacén temporal de velocidad 150 puede allanar las variaciones de velocidad en el vídeo comprimido. En algunas formas de realización, se emplea un modelo de almacén temporal de tasa de bits constante (CBR) en el que los bits se extraen del almacén temporal a una tasa de bits constante. En el modelo de almacén temporal de CBR, si el codificador de vídeo 20 agrega demasiados bits al flujo de bits, el almacén temporal de velocidad 150 puede desbordarse por lo alto. Por otro lado, el codificador de vídeo 20 debe agregar suficientes bits para evitar el desbordamiento por lo bajo del almacén temporal de velocidad 150.

[0042] En el lado del decodificador de vídeo, los bits se pueden agregar al almacén temporal de velocidad 155 del decodificador de vídeo 30 (véase la **FIG. 2B** que se describe con más detalle a continuación) a una tasa de bits constante, y el decodificador de vídeo 30 puede eliminar números variables de bits por cada bloque. Para garantizar la decodificación correcta, el almacén temporal de velocidad 155 del decodificador de vídeo 30 no debe "desbordarse por lo bajo" ["*underflow*"] o "desbordarse por lo alto" ["*overflow*"] durante la decodificación del flujo de bits comprimidos.

[0043] En algunas formas de realización, la plenitud del almacén temporal (BF) se puede definir en función de los valores de TamañoActualAlmacénTemporal, que representan el número de bits actualmente en el almacén temporal de velocidad 150, y los de TamañoMáxAlmacénTemporal, que representan el tamaño del almacén temporal de velocidad 150, es decir, el número máximo de bits que pueden ser almacenados en el almacén temporal de velocidad 150 en cualquier momento. La BF se puede calcular como:

$$BF = ((\text{TamañoActualAlmacénTemporal} * 100) / \text{TamañoMáxAlmacénTemporal}).$$

[0044] El detector de planicidad 115 puede detectar cambios desde áreas complejas (es decir, no planas) en los datos de vídeo a áreas planas (es decir, simples o uniformes) en los datos de vídeo. Los términos "complejo" y "plano" se usarán aquí para referirse generalmente a la dificultad del codificador de vídeo 20 para codificar las regiones respectivas de los datos de vídeo. Por lo tanto, el término "complejo", como se usa en este documento, generalmente describe una región de los datos de vídeo como compleja o difícil de codificar por el codificador de vídeo 20 y puede, por ejemplo, incluir datos de vídeo texturizados, alta frecuencia espacial y/u otras características que son complejas de codificar. Por ejemplo, se puede determinar que una región de los datos de vídeo es una región compleja cuando el número de bits necesarios para codificar la región es mayor que un umbral. El término "plano", como se usa en este documento, generalmente describe una región de los datos de vídeo como simple de codificar para el codificador de vídeo 20 y puede, por ejemplo, incluir un gradiente suave en los datos de vídeo, baja frecuencia espacial y/u otras características que son fáciles de codificar. Por ejemplo, se puede determinar que una región de los datos de vídeo es una región plana cuando el número de bits necesarios para codificar la región es menor que el umbral.

[0045] Sin embargo, según la implementación, la determinación de si una región dada es compleja o plana también puede determinarse en función de la norma de codificación utilizada, el hardware específico incluido en el codificador de vídeo 20, el tipo de datos de vídeo a codificar, etc. Además, ciertas propiedades de las regiones de datos de vídeo pueden influir en la cantidad de bits necesarios para codificar la región, por ejemplo, las regiones de alta textura y/o alta frecuencia espacial pueden requerir que se codifiquen más bits que las regiones de menor textura y/o menor frecuencia espacial. De manera similar, las regiones que comprenden ruido aleatorio pueden requerir un gran número de bits para ser codificadas, en comparación con las regiones más estructuradas de los datos de vídeo. Por lo tanto, en ciertas implementaciones, las regiones de los datos de vídeo se pueden identificar como regiones complejas y/o planas al comparar una medida de la textura y/o la frecuencia espacial (por ejemplo, un valor de complejidad) con un umbral de complejidad. Las transiciones entre regiones complejas y planas pueden ser utilizadas por el codificador de vídeo 20 para reducir las distorsiones de cuantización en los datos de vídeo codificados. Específicamente, el

controlador de frecuencia 120 y el componente predictor, cuantizador y reconstructor 125 pueden reducir tales distorsiones de cuantización cuando se identifican las transiciones de regiones complejas a planas. A continuación, se proporciona una descripción más detallada de la detección de planicidad y varias formas de realización de la misma.

5 **[0046]** El controlador de velocidad 120 determina un conjunto de parámetros de codificación, por ejemplo, un QP. El QP puede ser ajustado por el controlador de velocidad 120 basándose en la plenitud del almacén temporal 150 y la actividad de imagen de los datos de vídeo a fin de maximizar la calidad de la imagen para una tasa de bits de destino, lo que garantiza que el almacén temporal de la velocidad 150 no se desborda por lo alto o por lo bajo. El controlador de velocidad 120 también selecciona una opción de codificación particular (por ejemplo, una modalidad particular)
10 para cada bloque de los datos de vídeo a fin de lograr el rendimiento óptimo de velocidad-distorsión. El controlador de velocidad 120 minimiza la distorsión de las imágenes reconstruidas de manera que la distorsión satisfaga la restricción de velocidad de bits, es decir, la velocidad de codificación real global cabe dentro de la velocidad de bits de destino.

15 **[0047]** El componente predictor, cuantizador y reconstructor 125 puede realizar al menos tres operaciones de codificación del codificador de vídeo 20. El componente predictor, cuantizador y reconstructor 125 puede realizar la predicción en varias modalidades diferentes. Un ejemplo de modalidad de predicción es una versión modificada de la predicción adaptativa por mediana. La norma JPEG sin pérdida (JPEG-LS) puede implementar una predicción adaptativa a la mediana. La versión modificada de la predicción adaptativa por mediana que puede ser realizada por el componente predictor, cuantificador y reconstructor 125 puede admitir la predicción paralela de tres valores muestrales consecutivos. Otro ejemplo de modalidad de predicción es la predicción de bloques. En la predicción de bloques, las muestras se predicen a partir de los píxeles reconstruidos anteriormente a la izquierda. Tanto el codificador de vídeo 20 como el decodificador de vídeo 30 pueden realizar una búsqueda idéntica sobre píxeles reconstruidos para determinar los usos de predicción de bloques y, por lo tanto, no es necesario enviar bits en la modalidad de predicción de bloques. También se puede implementar una modalidad de predicción de punto medio en el que las muestras se predicen utilizando el punto medio del rango del componente. La modalidad de predicción de punto medio puede habilitar la acotación de la cantidad de bits necesarios para el vídeo comprimido, incluso en la muestra del caso más desfavorable.
20
25

30 **[0048]** El componente predictor, cuantizador y reconstructor 125 también realiza la cuantización. Por ejemplo, la cuantización se puede realizar mediante un cuantizador potencia de 2 que puede implementarse utilizando un desplazador. Se observa que se pueden implementar otras técnicas de cuantización en lugar del cuantizador potencia de 2. La cuantización realizada por el componente predictor, cuantizador y reconstructor 125 puede basarse en el QP determinado por el controlador de velocidad 120. Finalmente, el componente predictor, cuantizador y reconstructor 125 también realiza una reconstrucción que incluye agregar el residuo cuantizado inverso al valor predicho y asegurar que el resultado no caiga fuera del rango válido de valores de muestra.
35

40 **[0049]** Se observa que los enfoques a modo de ejemplo descritos anteriormente para la predicción, cuantización y reconstrucción realizadas por el componente predictor, cuantizador y reconstructor 125 son meramente ilustrativos y que se pueden implementar otros enfoques. También se señala que el componente predictor, cuantizador y reconstructor 125 puede incluir uno o más subcomponentes para realizar la predicción, la cuantización y/o la reconstrucción. Además, se señala que la predicción, la cuantización y/o la reconstrucción pueden realizarse mediante varios componentes codificadores por separado en lugar del componente predictor, cuantizador y reconstructor 125.

45 **[0050]** El almacén temporal de línea 130 contiene la salida del componente predictor, cuantizador y reconstructor 125, de modo que el componente predictor, cuantizador y reconstructor 125 y el historial cromático indizado 135 puedan utilizar los datos de vídeo almacenados temporalmente. El historial cromático indizado 135 almacena valores de píxeles utilizados recientemente. El codificador de vídeo 20 puede hacer referencia directamente a estos valores de píxeles utilizados recientemente mediante una sintaxis dedicada.

50 **[0051]** El codificador por entropía 140 codifica los residuos de predicción recibidos desde el componente predictor, cuantizador y reconstructor 125 basándose en el historial cromático indizado 135 y las transiciones de planicidad identificadas por el detector de planicidad 115. En algunos ejemplos, el codificador por entropía 140 puede codificar tres muestras por reloj por cada codificador de subflujo. El multiplexor de subflujo 145 puede multiplexar el flujo de bits basándose en un esquema de multiplexado de paquetes sin encabezado. Esto permite que el decodificador de vídeo 30 ejecute tres decodificadores por entropía en paralelo, facilitando la decodificación de tres píxeles por reloj. El multiplexor de subflujo 145 puede optimizar el orden de los paquetes de modo que el decodificador de vídeo 30 pueda decodificar los paquetes de manera eficaz. Se observa que se pueden implementar diferentes enfoques para la codificación por entropía, lo que puede facilitar la decodificación de una potencia de 2 píxeles por reloj (por ejemplo, 2 píxeles/reloj o 4 píxeles/reloj).
55
60

Decodificador de Vídeo DSC

65 **[0052]** La FIG. 2B es un diagrama de bloques que ilustra un ejemplo del decodificador de vídeo 30 que puede implementar las técnicas de acuerdo con aspectos descritos en esta divulgación. El decodificador de vídeo 30 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del codificador de vídeo 30. En

algunos ejemplos, adicionalmente o de forma alternativa, un procesador (no mostrado) puede configurarse para realizar algunas de, o todas, las técnicas descritas en esta divulgación.

5 [0053] Con fines de explicación, esta divulgación describe el decodificador de vídeo 30 en el contexto de la codificación por DSC. Sin embargo, las técnicas de esta divulgación pueden ser aplicables a otras normas o procedimientos de codificación.

10 [0054] En el ejemplo de la **FIG. 2B**, el decodificador de vídeo 30 incluye una pluralidad de componentes funcionales. Los componentes funcionales del decodificador de vídeo 30 incluyen un almacén temporal de velocidad 155, un demultiplexor de subflujo 160, un decodificador por entropía 165, un controlador de velocidad 170, un componente predictor, cuantizador y reconstructor 175, un historial cromático indizado 180, un almacén temporal de línea 185 y un convertidor de espacio cromático 190. Los componentes ilustrados del decodificador de vídeo 30 son análogos a los componentes correspondientes descritos anteriormente en relación con el codificador de vídeo 20 en la **FIG. 2A**. Como tal, cada uno de los componentes del decodificador de vídeo 30 puede funcionar de una manera similar a los componentes correspondientes del codificador de vídeo 20, como se ha descrito anteriormente.

Fragmentos en DSC

20 [0055] Como se ha señalado anteriormente, un fragmento generalmente se refiere a una región espacialmente distinta en una imagen o una trama que puede decodificarse independientemente sin usar la información del resto de las regiones en la imagen o trama. Cada imagen o trama de vídeo se puede codificar en un solo fragmento o la imagen o la trama de vídeo se puede codificar en varios fragmentos. En DSC, los bits de destino asignados para codificar cada fragmento pueden ser esencialmente constantes.

25 Detección de Planicidad

[0056] Una técnica para la detección de planicidad puede implicar identificar la transición desde una región compleja a una región plana. En algunas implementaciones, el algoritmo de control de velocidad, que puede ser realizado por el controlador de velocidad 120, incluye la selección de un valor de QP para un bloque actual basándose, al menos en parte, en una velocidad de bits de un bloque anterior. Cuando la velocidad de bits del bloque anterior es más alta que una velocidad media de bits de los bloques codificados previamente (por ejemplo, debido a una alta actividad), el algoritmo de control de velocidad aumenta el valor de QP del bloque actual para mantener la restricción de velocidad de bits y también para evitar el desbordamiento por lo alto del almacén temporal de velocidad 150. Por lo tanto, el valor de QP del bloque actual puede alcanzar un valor muy alto, cuando una serie de bloques consecutivos codificados previamente son sumamente complejos. Cuando una región plana sigue inmediatamente después de una región compleja, las distorsiones visuales pueden notarse si la región plana está codificada con un QP alto. Por lo tanto, puede ser deseable identificar esta transición desde una región compleja de datos de vídeo a una región plana de datos de vídeo para que la codificación de la región plana se pueda realizar utilizando un valor de QP bajo.

40 [0057] En algunas implementaciones, la determinación de la planicidad se realiza para cada grupo o bloque (por ejemplo, 3 píxeles) dentro de un supergrupo (por ejemplo, 4 grupos/bloques consecutivos). Se pueden realizar dos verificaciones de planicidad para determinar el tipo de planicidad, que puede ser "no plana", "algo plana" o "muy plana". Por lo tanto, para cada supergrupo que tenga un tamaño de 4 grupos, como en el ejemplo anterior, puede haber cuatro indicaciones de planicidad, una para cada grupo/bloque. Cada indicación de planicidad puede indicar, por ejemplo, que el bloque-grupo correspondiente es no plano, algo plano o muy plano. En algunas formas de realización, se puede determinar que un bloque algo plano es una región de los datos de vídeo que es más compleja o difícil de codificar que una región muy plana y más simple de codificar que una región no plana. Por ejemplo, se puede determinar que una región de los datos de vídeo es una región algo plana cuando una representación de la textura y/o la frecuencia espacial (por ejemplo, un valor de complejidad) es mayor que un primer umbral y menor que un segundo umbral.

50 [0058] La **FIG. 3** ilustra los píxeles implicados en la determinación de planicidad para un grupo actual de píxeles según un ejemplo. Haciendo referencia a la **FIG. 3**, se muestra una pluralidad de píxeles 205 a 235. Un grupo actual de píxeles se muestra en la **FIG. 3**, que incluye los píxeles 210, 215 y 220. El píxel 205 es un píxel que se encuentra antes del grupo actual. En otras palabras, el píxel 205 constituye un píxel anterior. Los píxeles 225, 230 y 235 se encuentran después del grupo actual. En otras palabras, los píxeles 225, 230 y 235 constituyen los píxeles siguientes.

60 [0059] La primera verificación de planicidad puede incluir dos pruebas. En la primera prueba para el tipo muy plano de planicidad, los valores mínimo y máximo para cada componente de color se determinan entre las cuatro muestras, incluyendo el píxel anterior 205 y los píxeles 210, 215 y 220 del grupo actual. Si la diferencia entre el valor máximo y el valor mínimo de cualquier componente de color es mayor que $(2 \ll (\text{bitsPorComponente}-8))$, donde \ll es el operador de desplazamiento a la izquierda a nivel de bits, entonces la primera prueba para una región muy plana falla. Si la diferencia entre el valor máximo y el valor mínimo de cualquier componente de color no es mayor que $(2 \ll (\text{bitsPorComponente}-8))$, entonces la primera prueba se aprueba, por ejemplo, se determina que el grupo actual de píxeles tiene un valor muy plano de planicidad.

65

[0060] En algunas formas de realización, cuando se aprueba la primera prueba, la segunda prueba no se realiza. En otras formas de realización, la segunda prueba se realiza independientemente de si se aprueba la primera prueba, o cuando la primera prueba falla. En la segunda prueba para el tipo algo plano de planicidad, si la diferencia entre el valor máximo y el valor mínimo de cualquier componente de color entre las cuatro muestras, incluyendo el píxel anterior 205 y los píxeles 210, 215 y 220, es mayor que un umbral, entonces la segunda prueba para regiones algo planas falla. Si la diferencia entre el valor máximo y el valor mínimo de cualquier componente de color no es mayor que el umbral, entonces se aprueba la segunda prueba, por ejemplo, se determina que el grupo actual de píxeles tiene el tipo de planicidad algo plana. En ciertas implementaciones, el umbral utilizado en la segunda prueba se calcula en función del QP.

[0061] Si las primeras y segundas pruebas descritas anteriormente fallan en la primera verificación de planicidad, entonces la segunda verificación de planicidad se realiza sobre los píxeles en el grupo actual y los tres píxeles siguientes (es decir, para los píxeles 210 a 235). Durante la segunda verificación de planicidad, se calculan los valores mínimos y máximos de los píxeles 210 a 235 y se realizan las mismas dos pruebas realizadas en la primera verificación de planicidad para los píxeles 210 a 235.

[0062] Finalmente, cada grupo actual, por ejemplo, los píxeles 210, 215 y 220, en el supergrupo se clasifica como plano, algo plano o no plano según el siguiente procedimiento mostrado en pseudocódigo:

```

Repetir para cuatro grupos en supergrupo {
    si (!prevIsFlat && grupo es muy plano o algo plano)
        Se señala el grupo actual y el tipo de planicidad.
    en caso contrario
        prevIsFlat == 0
}

```

[0063] El valor de prevIsFlat se inicializa en 1 si el supergrupo anterior tiene una indicación de planicidad; de lo contrario, el valor de prevIsFlat se inicializa en 0. Si se selecciona un grupo dentro del supergrupo que sea plano o algo plano, un indicador (designado como indicador_planicidad) para el supergrupo se fija en 1 y el grupo correspondiente se señala en el flujo de bits junto con el tipo de planicidad asociado. Si no se selecciona ningún grupo que sea plano o algo plano, el indicador_planicidad para el supergrupo se fija en 0. Finalmente, el controlador de velocidad 120 puede ajustar el valor del QP del grupo actual si se encuentra que el grupo actual es plano o algo plano.

[0064] Puede haber una serie de limitaciones asociadas a los procedimientos de detección de planicidad descritos anteriormente. Por ejemplo, estos procedimientos pueden requerir la señalización explícita de si el bloque o grupo actual es plano o no plano y pueden requerir además la señalización del tipo de planicidad, es decir, muy plano o algo plano, cuando el bloque o grupo actual es plano. Los bits requeridos para esta señalización explícita pueden limitar la compresión alcanzable al implementar los procedimientos de detección de planicidad descritos anteriormente.

[0065] Además, el procedimiento de detección de planicidad descrito anteriormente puede no ajustarse a escala para la detección de planicidad de grandes tamaños de grupo o bloque (es decir, cuando el número de píxeles en el grupo/bloque es relativamente grande). Esto se debe a que los valores mínimos y máximos para grupos o bloques más grandes pueden representar valores más extremos y pueden no lograr capturar las variaciones espaciales locales dentro del grupo.

[0066] La **FIG. 4** es un diagrama de bloques que ilustra el detector de planicidad 115 de acuerdo con una forma de realización. El detector de planicidad 115 puede incluir tres calculadoras de complejidad 332, 324 y 326, una calculadora de máximos 330 y un detector de transición 340. Las calculadoras de complejidad 332, 324 y 326 calculan respectivamente las complejidades de los bloques correspondientes. Los valores de complejidad calculados por las calculadoras de complejidad 332, 324 y 326 pueden ser respectivamente representativos de la textura y/o la frecuencia espacial de los bloques correspondientes. El detector de planicidad 115 puede determinar si se produce una transición desde una región no plana a una región plana en un bloque actual 310 basándose en un bloque anterior 305 y un bloque siguiente 315. Aunque el detector de planicidad 115 se ha ilustrado como incluyendo una serie de bloques independientes, según la forma de realización, los bloques pueden combinarse y/o implementarse entre sí o en varias combinaciones.

[0067] Como se muestra en la **FIG. 4**, las complejidades de los bloques actual, siguiente y anterior 305, 310 y 315 se calculan mediante las respectivas calculadoras de complejidad 322, 324 y 326. Las complejidades de los bloques actual, siguiente y anterior 305, 310 y 315 se indicarán respectivamente como C_{act} , C_{sig} , C_{ant} de aquí en adelante. La calculadora de máximos 330 calcula la complejidad máxima, C_{max} , que se ha identificado hasta ahora en la línea o fragmento actual. El detector de transición 340 identifica la transición desde una región compleja a una región plana basándose en los valores de complejidad calculados C_{act} , C_{sig} , C_{ant} y C_{max} . En ciertas implementaciones, mientras se codifica el primer bloque en una línea o fragmento, C_{max} se inicializa con el máximo entre C_{act} y C_{sig} . Después de codificar todos y cada uno de los bloques, el valor de C_{max} se actualiza en función del máximo entre C_{max} y C_{sig} . Finalmente, el valor de QP del bloque actual 310 se calcula en función, al menos en parte, de si ha sido identificada

una transición desde una región compleja a una región plana en el bloque actual 310 por el detector de transición 340. En algunas implementaciones, el valor de QP se fija en un valor relativamente bajo cuando se ha identificado una transición desde una región compleja a una región plana en el bloque actual 310.

5 **[0068]** En una implementación, la calculadora máxima 330 calcula el valor de C_{max} de forma independiente para cada línea en un fragmento. Es decir, para el primer bloque en cada nueva línea, la calculadora de máximos 330 inicializa C_{max} con el máximo entre C_{act} y C_{sig} y además actualiza C_{max} basándose en el máximo entre C_{max} y C_{sig} .

10 **[0069]** El cálculo de complejidad para un bloque seleccionado (es decir, el cálculo realizado por las calculadoras de complejidad 322, 324 y 326 para los respectivos bloques anterior, actual y siguiente 305, 310 y 315) de acuerdo con formas de realización a modo de ejemplo se describirá ahora con más detalle. El valor de complejidad del bloque seleccionado se calcula aplicando primero una transformación específica al bloque seleccionado y luego sumando los valores absolutos o valores cuadrados absolutos de los coeficientes de transformación. En ciertas implementaciones, solo el canal de luma se usa para calcular el valor de complejidad y, en otras implementaciones, tanto el canal de luma como el de croma se usan para calcular el valor de complejidad. En una forma de realización a modo de ejemplo, la transformación aplicada al bloque seleccionado es una transformación de coseno discreta (DCT) o transformación de Hadamard. En otro ejemplo, el valor de complejidad puede obtenerse en función del número de bits utilizados para codificar los bloques y el valor de QP respectivo. En otro ejemplo, el valor de complejidad puede obtenerse basándose en la energía residual entre el bloque original y un bloque predicho.

15 **[0070]** En otra forma de realización a modo de ejemplo, se utiliza un subconjunto de los coeficientes de transformación generados por la aplicación de la transformación al bloque seleccionado para calcular la suma absoluta o la suma cuadrada absoluta. En esta forma de realización, no todos los coeficientes de transformación en el bloque se utilizan para calcular la suma absoluta o la suma cuadrada absoluta.

20 **[0071]** En otra forma de realización a modo de ejemplo, cada coeficiente de transformación se multiplica por una ponderación y las ponderaciones aplicadas a los coeficientes de transformación pueden ser diferentes entre sí. En esta realización, se calcula el valor absoluto o el valor cuadrado absoluto de los coeficientes ponderados.

25 **[0072]** En otra forma de realización a modo de ejemplo más, se aplica una transformación de color al bloque antes de calcular los coeficientes de transformación.

30 **[0073]** En otra forma de realización más, para cada canal de color, se suman los valores absolutos o cuadrados del subconjunto de los respectivos coeficientes de transformación y la suma final se normaliza. En esta forma de realización, las sumas normalizadas de todos los canales de color se suman para generar el valor de complejidad. En una implementación, la normalización incluye dividir el valor acumulado por el número de píxeles en el bloque seleccionado. En otra implementación, antes de dividir el valor acumulado por el número de píxeles en el bloque seleccionado, se agrega un valor de compensación al valor acumulado para redondear el valor acumulado al número entero más cercano. Por ejemplo, el valor de compensación puede ser el número de píxeles en el bloque dividido por 2. Se observa que una o más de las formas de realización o enfoques descritos anteriormente para los cálculos de complejidad pueden combinarse.

35 **[0074]** En otra forma de realización, para cada canal de color, se suman los valores absolutos o cuadrados del subconjunto de los respectivos coeficientes de transformación y la suma final se normaliza. La suma normalizada de cada canal de color se puede multiplicar por una ponderación correspondiente. Las sumas normalizadas ponderadas pueden luego sumarse entre sí para obtener el valor de complejidad. En una implementación, los datos de vídeo de entrada están en el espacio cromático RGB y el convertidor de espacio cromático 105 convierte los datos de vídeo de entrada en el espacio cromático YCgCo. En esta implementación, la ponderación para el canal Y puede ser, por ejemplo, 1 y cada una de las ponderaciones para los canales de croma C_o y C_g puede ser, por ejemplo, 0.5.

40 **[0075]** Como se describe más arriba, el detector de transición 340 identifica si se ha producido una transición desde una región compleja a una región plana en el bloque 310 actual. En ciertas formas de realización, la transición desde una región compleja a una región plana es identificada por el detector de transición 340, basándose, al menos en parte, en: (i) una primera condición que se cumple cuando el siguiente bloque 315 es plano; y (ii) una segunda condición que se cumple cuando el bloque anterior 305 es complejo. A continuación, se describen detalladamente varias implementaciones que ilustran cuándo se cumplen las condiciones (i) y (ii). Cuando se cumplen ambas condiciones (i) y (ii), el controlador de velocidad 120 ajusta el valor de QP del bloque actual 310 en un valor relativamente bajo. De lo contrario, si una cualquiera de las condiciones falla, el controlador de velocidad 120 no ajusta el valor de QP del bloque actual 310 basándose en la salida del detector de planicidad 115.

45 **[0076]** En formas de realización a modo de ejemplo, el detector de transición 340 determinará que la condición (i) se cumple si la siguiente condición es verdadera:

$$((C_{Act} - C_{sig}) > C_{max} * T_1) \ \&\& \ (C_{sig} < T_2 * C_{max}),$$

50 donde T_1 y T_2 son parámetros de umbral que son configurables.

[0077] En una implementación, la verificación condicional de 'mayor que' en $((C_{Act} - C_{sig}) > C_{max} * T_1)$ se reemplaza con una verificación condicional de 'mayor o igual a', es decir, $((C_{Act} - C_{sig}) \geq C_{max} * T_1) \ \&\& \ (C_{sig} < T_2 * C_{max})$.

5 **[0078]** En otra implementación, la verificación condicional de 'menor que' en $(C_{sig} < T_2 * C_{max})$ se reemplaza con una verificación condicional de 'menor o igual a', por ejemplo, $(C_{sig} \leq T_2 * C_{max})$.

[0079] En otra implementación, se usa la siguiente condición $((C_{Act} - C_{sig}) \geq C_{max} * T_1) \ \&\& \ (C_{sig} \leq T_2 * C_{max})$ para determinar si se cumple la condición (i).

10 **[0080]** En otra forma de realización a modo de ejemplo, la condición (i) se cumple si la siguiente condición es verdadera:

$$C_{sig} < C_{Act} * T,$$

15 donde T es parámetro de umbral configurable.

[0081] En una implementación, la verificación condicional $C_{sig} \leq C_{Act} * T$ se usa en lugar de la verificación condicional anterior para ver si se cumple la condición (i).

20 **[0082]** En otra forma de realización a modo de ejemplo más, la condición (ii) se cumple si la siguiente condición es verdadera:

$$(C_{ant} > C_{max} * T_3),$$

25 donde T_3 es un parámetro de umbral que es configurable.

[0083] En una implementación, la verificación condicional de 'mayor que' en $(C_{ant} > C_{max} * T_3)$ se reemplaza con una verificación condicional de 'mayor o igual a', es decir, $(C_{ant} \geq C_{max} * T_3)$.

30 **[0084]** En otra forma de realización a modo de ejemplo, la condición (ii) se cumple si $C_{Ant} > T_4$, donde T_4 es un parámetro configurable.

[0085] En una implementación, la verificación condicional de 'mayor que' $C_{ant} > T_4$ se reemplaza por una verificación condicional de 'mayor o igual', es decir, $C_{ant} \geq T_4$.

35 **[0086]** En otra forma de realización más, la condición (ii) se evalúa basándose en el siguiente algoritmo:

```

40     para j = 1 a n
        {
            si  $(C_{sig} \leq A[j])$ 
                {
45                 si  $(C_{ant} > B[j])$ 
                    {
                        se cumple la condición (ii);
                        corte;
                    }
                }
        }
50     }

```

[0087] Aquí, $A = [a_1, a_2, \dots, a_n]$, $B = [b_1, b_2, \dots, b_n]$, donde a_j y b_j son valores de umbral configurables, para todos los valores de j desde 1 hasta n. Además, 'n' representa un entero positivo. En el algoritmo anterior, tan pronto como se encuentra que se cumple la condición (ii), el algoritmo termina mediante el corte.

55 **[0088]** En una implementación, n se fija en 1 y, por lo tanto, se utiliza un valor de umbral en cada uno entre A y B.

[0089] En una implementación, la verificación condicional de 'menor o igual a' $(C_{sig} \leq A[j])$ se reemplaza con una verificación condicional de 'menor que', por ejemplo, $(C_{sig} < A[j])$. En otra implementación, la verificación condicional de 'mayor que' $C_{ant} > B[j]$ se reemplaza por una verificación condicional de 'mayor o igual a', es decir, $C_{ant} \geq B[j]$.

60 **[0090]** En una implementación, los valores de umbral a_1, a_2, \dots, a_n están en orden monótonamente creciente. En una implementación, los valores de umbral b_1, b_2, \dots, b_n están en orden monótonamente creciente.

[0091] En otra implementación, la verificación condicional $C_{ant} > T_4$ y el algoritmo anterior se combinan para determinar si se cumple la condición (ii).

5 [0092] En una implementación, las etapas en el algoritmo anterior se llevan a cabo al principio para verificar si se cumple la condición (ii). Si la condición (ii) no se cumple después de completar las etapas en este algoritmo, se lleva a cabo la verificación condicional $C_{ant} > T_4$.

10 [0093] En una implementación, las condiciones (i) y (ii) se verifican si el valor C_{max} es mayor, o mayor o igual, que un valor umbral T_4 que es configurable. Es decir, si se cumple la condición ($C_{max} \geq T_4$), se comprueban las condiciones (i) y (ii) para detectar la transición desde una región no plana a una región plana. De lo contrario, si la condición ($C_{max} \geq T_4$) no se cumple, ambas condiciones (i) y (ii) no se verifican y el valor de QP no se ajusta. En una implementación, la verificación condicional de 'mayor o igual a' se reemplaza con una verificación condicional de 'estrictamente mayor que', es decir, ($C_{max} > T_4$).

15 **Ejemplo de Diagrama de Flujo para Identificar una Transición desde una Región Compleja hasta una Región Plana en datos de Vídeo**

[0094] Con referencia a la FIG. 5, se describirá un ejemplo de procedimiento para identificar una transición desde una región compleja a una región plana en datos de vídeo. La FIG. 5 es un diagrama de flujo que ilustra un procedimiento 400 para codificar datos de vídeo, de acuerdo con una forma de realización de la presente divulgación. Las etapas ilustradas en la FIG. 5 pueden ser realizadas por un codificador de vídeo (por ejemplo, el codificador de vídeo 20 en la FIG. 2A) o uno o varios componentes del mismo. Por conveniencia, el procedimiento 400 se describe como realizado por un codificador de vídeo (también denominado simplemente codificador), que puede ser el codificador de vídeo 20 u otro componente.

25 [0095] El procedimiento 400 comienza en el bloque 401. En el bloque 405, el codificador determina un primer valor de complejidad para un bloque anterior de los datos de vídeo, un segundo valor de complejidad para un bloque actual de los datos de vídeo y un tercer valor de complejidad para un bloque siguiente de los datos de vídeo. Los valores de complejidad primero, segundo y tercero son representativos de una textura y/o una frecuencia espacial del bloque correspondiente entre el bloque anterior, el bloque actual y los bloques siguientes de los datos de vídeo. En algunas formas de realización, cada uno de los valores de complejidad primero, segundo y tercero puede calcularse generando una pluralidad de coeficientes de transformación mediante la aplicación de una transformación al bloque correspondiente de los bloques anterior, actual y siguiente. En estas formas de realización, cada uno de los valores de complejidad correspondientes a los bloques anterior, actual y siguiente se determina sumando los valores absolutos o los valores cuadrados absolutos de los coeficientes de transformación.

30 [0096] En el bloque 410, el codificador determina si el bloque actual incluye una transición desde una región compleja a una región plana, en función de los valores de complejidad primero, segundo y tercero. La región compleja tiene un valor de complejidad que cumple o supera un umbral de complejidad y la región plana tiene un valor de complejidad que es menor que el umbral de complejidad. En algunas formas de realización, el codificador determina además un valor de complejidad máxima para un fragmento actual de los datos de vídeo. El codificador también puede determinar si se cumple una primera condición, basándose, al menos en parte, en el tercer valor de complejidad y determinar si se cumple una segunda condición basándose, al menos en parte, en el primer valor de complejidad, para determinar si el bloque actual incluye una transición desde una región compleja. En el bloque 415, el codificador codifica los datos de vídeo basándose, al menos en parte, en la determinación de si el bloque actual incluye la transición desde la región compleja a la región plana. El procedimiento termina en el bloque 420.

35 [0097] En el procedimiento 400, uno o más de los bloques mostrados en la FIG. 5 pueden eliminarse (por ejemplo, no realizarse) y/o puede conmutarse el orden en que se realiza el procedimiento. En algunas formas de realización, se pueden agregar bloques adicionales al procedimiento 400. Las formas de realización de la presente divulgación no están limitadas a, o por, el ejemplo mostrado en la FIG. 5, y otras variaciones pueden implementarse sin apartarse de esta divulgación.

40 **Otras Consideraciones**

55 [0098] La información y las señales divulgadas en el presente documento pueden representarse usando cualquiera entre varias tecnologías y técnicas diferentes. Por ejemplo, los datos, las instrucciones, los comandos, la información, las señales, los bits, los símbolos y los chips que puedan haberse mencionado a lo largo de la descripción anterior pueden representarse mediante tensiones, corrientes, ondas electromagnéticas, campos o partículas magnéticos, campos o partículas ópticos o cualquier combinación de los mismos.

60 [0099] Los diversos bloques lógicos y etapas de algoritmo ilustrativos descritos en relación con las formas de realización divulgadas en el presente documento pueden implementarse como hardware electrónico, software informático o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, anteriormente se han descrito, en general, diversos componentes, bloques y etapas ilustrativas en términos de su funcionalidad. Si dicha funcionalidad se implementa como hardware o software depende de la aplicación y las

restricciones de diseño particulares impuestas al sistema global. Los expertos en la técnica pueden implementar la funcionalidad descrita de varias maneras para cada aplicación particular, pero no se debería interpretar que dichas decisiones de implementación suponen apartarse del alcance de la presente divulgación.

5 **[0100]** Las técnicas descritas en el presente documento pueden implementarse en hardware, software, firmware o en cualquier combinación de lo anterior. Dichas técnicas pueden implementarse en cualquiera entre una variedad de dispositivos tales como ordenadores de propósito general, equipos manuales de dispositivos de comunicación inalámbrica o dispositivos de circuitos integrados que tienen múltiples usos, incluyendo su aplicación en equipos manuales de dispositivos de comunicación inalámbrica y otros dispositivos. Cualquier característica descrita como
 10 dispositivos o componentes pueden implementarse juntas en un dispositivo lógico integrado o por separado, como dispositivos lógicos discretos pero interoperables. Si se implementan en software, las técnicas pueden realizarse, al menos en parte, mediante un medio de almacenamiento de datos legible por ordenador que comprenda código de programa que incluye instrucciones que, cuando se ejecutan, realizan uno o más de los procedimientos descritos anteriormente. El medio de almacenamiento de datos legible por ordenador puede formar parte de un producto de
 15 programa informático, que puede incluir materiales de embalaje. El medio legible por ordenador puede comprender memoria o medios de almacenamiento de datos, tales como memoria de acceso aleatorio (RAM), tal como memoria de acceso aleatorio dinámica síncrona (SDRAM), memoria de sólo lectura (ROM), memoria de acceso aleatorio no volátil (NVRAM), memoria de sólo lectura programable y borrable eléctricamente (EEPROM), memoria FLASH, medios de almacenamiento de datos magnéticos u ópticos, y similares. Las técnicas se pueden realizar, adicionalmente o de
 20 forma alternativa, al menos en parte, por un medio de comunicación legible por ordenador que lleva o comunica código de programa en forma de instrucciones o estructuras de datos y a las que se puede acceder, leer y/o ejecutar por medio de un ordenador, tales como señales u ondas propagadas.

[0101] El código de programa puede ser ejecutado por un procesador, que puede incluir uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables en el terreno (FPGA) u otros circuitos lógicos equivalentes, integrados o discretos. Un procesador de este tipo puede estar configurado para realizar cualquiera de las técnicas descritas en esta divulgación. Un procesador de propósito general puede ser un microprocesador, pero, como alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador o máquina de estados convencional. Un procesador también puede implementarse como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores junto con un núcleo de DSP o cualquier otra configuración de este tipo. Por consiguiente, el término "procesador", como se usa en el presente documento, puede referirse a cualquier estructura anterior, cualquier combinación de la estructura anterior, o cualquier otra estructura o aparato adecuados 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 software o hardware dedicados configurados para la codificación y la decodificación, o incorporados en un codificador-decodificador de vídeo combinado (CODEC). Asimismo, las técnicas podrían implementarse por completo en uno o más circuitos o elementos lógicos.

40 **[0102]** Las técnicas de la presente divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un equipo manual inalámbrico, un circuito integrado (CI) o un conjunto de CI (por ejemplo, un conjunto de chips). Varios componentes o unidades se describen en esta divulgación para enfatizar los aspectos funcionales de los dispositivos configurados para realizar las técnicas divulgadas, pero no necesariamente requieren la realización por diferentes unidades de hardware. Más bien, como se ha descrito anteriormente, diversas unidades pueden combinarse en una unidad de hardware de códec o proporcionarse por una colección de unidades de hardware interoperativas, incluyendo uno o más procesadores como se describe anteriormente, junto con el software y/o firmware adecuados.

50 **[0103]** Aunque lo anterior ha sido descrito en relación con diversas formas de realización diferentes, las características o elementos de una forma de realización se pueden combinar con otras formas de realización sin apartarse de las enseñanzas de esta divulgación. Sin embargo, las combinaciones de características entre las formas de realización respectivas no están necesariamente limitadas a ellas. Se han descrito diversas formas de realización de la divulgación. Estas y otras formas de realización están dentro del alcance de las siguientes reivindicaciones.

55

REIVINDICACIONES

1. Un procedimiento para codificar datos de vídeo, que comprende:

5 determinar un primer valor de complejidad para un bloque anterior de los datos de vídeo, un segundo valor de complejidad para un bloque actual de los datos de vídeo y un tercer valor de complejidad para un bloque siguiente de los datos de vídeo, siendo los valores de complejidad primero, segundo y tercero representativos de al menos una de entre una textura o una frecuencia espacial del bloque correspondiente de entre el bloque anterior, el bloque actual y el bloque siguiente de los datos de vídeo, siendo los bloques anterior, actual y siguiente de los datos de vídeo consecutivos en orden de codificación y estando ubicados en un mismo fragmento;

10 determinar si el bloque actual incluye una transición desde una región compleja hasta una región plana en base, al menos en parte, a los valores de complejidad primero, segundo y tercero, en donde las regiones complejas tienen valores de complejidad que cumplen o superan un umbral de complejidad y las regiones planas tienen valores de complejidad que son menores que el umbral de complejidad;

15 calcular un valor de parámetro de cuantificación (QP) para el bloque actual en base, al menos en parte, a la determinación de si el bloque actual incluye una transición desde una región compleja hasta una región plana; y

20 codificar los datos de vídeo en base, al menos en parte, al valor de QP calculado.

2. El procedimiento de la reivindicación 1, en el que la determinación de los valores de complejidad primero, segundo y tercero comprende:

25 seleccionar uno de entre los bloques anterior, actual y siguiente;

30 calcular una pluralidad de coeficientes de transformación mediante la aplicación de una transformación al bloque seleccionado; y

35 determinar el valor de complejidad correspondiente al bloque seleccionado mediante la suma de los valores absolutos o los valores cuadrados absolutos de los coeficientes de transformación.

3. El procedimiento de la reivindicación 2, en el que la determinación de los valores de complejidad primero, segundo y tercero comprende además aplicar una transformación cromática al bloque seleccionado antes de calcular la pluralidad de coeficientes de transformación.

4. El procedimiento de la reivindicación 1, en el que la determinación del valor de complejidad para cada uno de los bloques anterior, actual y siguiente comprende:

35 calcular una pluralidad de coeficientes de transformación mediante la aplicación de una transformación a un bloque correspondiente;

40 sumar el valor absoluto o el valor cuadrado absoluto de un subconjunto de los coeficientes de transformación para cada canal de color;

45 normalizar la suma del valor absoluto o del valor cuadrado absoluto para cada canal de color;

50 aplicar una ponderación correspondiente a la suma normalizada del valor absoluto o del valor cuadrado absoluto para cada canal de color; y

55 determinar el valor de complejidad del bloque correspondiente mediante la suma de la suma normalizada ponderada para cada uno de los canales de color.

5. El procedimiento de la reivindicación 1, que comprende además determinar un valor de complejidad máximo para un fragmento actual de los datos de vídeo, en el que determinar si el bloque actual incluye la transición desde la región compleja hasta la región plana se basa además en el valor de complejidad máximo.

6. El procedimiento de la reivindicación 5, en el que la determinación del valor de complejidad máximo para el fragmento actual de los datos de vídeo comprende:

55 cuando el bloque anterior es un primer bloque en el fragmento actual y el bloque actual es un segundo bloque en el fragmento actual, inicializar el valor de complejidad máximo a un valor máximo entre los valores de complejidad primero y segundo; y

60 actualizar el valor de complejidad máximo a un valor máximo entre el valor de complejidad máximo y el valor de complejidad tercero.

7. El procedimiento de la reivindicación 1, en el que la determinación de si el bloque actual incluye la transición desde la región compleja hasta la región plana comprende:

65 determinar si se cumple una primera condición en base, al menos en parte, al valor de complejidad tercero para el bloque siguiente; y

determinar si se cumple una segunda condición en base, al menos en parte, al valor de complejidad primero para el bloque anterior.

8. El procedimiento de la reivindicación 7, en el que la primera condición se cumple cuando la siguiente condición es verdadera:

$$((C_{Act} - C_{sig}) > C_{max} * T_1) \&\& (C_{sig} < T_2 * C_{max}),$$

donde C_{Act} es el valor de complejidad segundo, C_{sig} es el valor de complejidad tercero, C_{max} es el valor de complejidad máximo, y T_1 y T_2 son los valores de umbral primero y segundo.

9. El procedimiento de la reivindicación 7, en el que la determinación de si se cumple la segunda condición comprende:

determinar si el valor de complejidad tercero es menor o igual a un valor de umbral tercero;
 determinar si el valor de complejidad primero es mayor que un cuarto valor de umbral; y
 determinar que la segunda condición se cumple cuando el valor de complejidad tercero es menor o igual al valor de umbral tercero y el valor de complejidad primero es mayor que el cuarto valor de umbral.

10. Un aparato, que comprende:

medios para determinar un primer valor de complejidad para un bloque anterior de datos de vídeo, un segundo valor de complejidad para un bloque actual de los datos de vídeo y un tercer valor de complejidad para un bloque siguiente de los datos de vídeo, siendo los valores de complejidad primero, segundo y tercero representativos de al menos una de entre una textura o una frecuencia espacial del bloque correspondiente de entre el bloque anterior, el bloque actual y el bloque siguiente de los datos de vídeo, siendo los bloques anterior, actual y siguiente de los datos de vídeo consecutivos en orden de codificación y estando ubicados en un mismo fragmento;

medios para determinar si el bloque actual incluye una transición desde una región compleja hasta una región plana en base, al menos en parte, a los valores de complejidad primero, segundo y tercero, en donde las regiones complejas tienen valores de complejidad que cumplen o superan un umbral de complejidad y las regiones planas tienen valores de complejidad que son menores que el umbral de complejidad;

medios para calcular un valor de parámetro de cuantización (QP) para el bloque actual en base, al menos en parte, a la determinación de si el bloque actual incluye una transición desde una región compleja hasta una región plana; y

medios para codificar los datos de vídeo en base, al menos en parte, al valor de QP calculado.

11. El aparato de la reivindicación 10, en el que los medios para determinar los valores de complejidad primero, segundo y tercero comprenden:

medios para seleccionar uno de los bloques anterior, actual y siguiente;

medios para calcular una pluralidad de coeficientes de transformación mediante la aplicación de una transformación al bloque seleccionado; y

medios para determinar el valor de complejidad correspondiente al bloque seleccionado mediante la suma de los valores absolutos o los valores cuadrados absolutos de los coeficientes de transformación.

12. El aparato de la reivindicación 10, en el que los medios para determinar el valor de complejidad para cada uno de los bloques anterior, actual y siguiente comprenden:

medios para calcular una pluralidad de coeficientes de transformación mediante la aplicación de una transformación a un bloque correspondiente;

medios para sumar el valor absoluto o el valor cuadrado absoluto de un subconjunto de los coeficientes de transformación para cada canal de color;

medios para normalizar la suma del valor absoluto o del valor cuadrado absoluto para cada canal de color;

medios para aplicar una ponderación correspondiente a la suma normalizada del valor absoluto o del valor cuadrado absoluto para cada canal de color; y

medios para determinar el valor de complejidad del bloque correspondiente mediante la suma de la suma normalizada ponderada para cada uno de los canales de color.

13. El aparato de la reivindicación 10, que comprende además medios para determinar un valor de complejidad máximo para un fragmento actual de los datos de vídeo, en el que determinar si el bloque actual incluye la transición desde la región compleja hasta la región plana se basa además en el valor de complejidad máximo.

14. El aparato de la reivindicación 13, en el que los medios para determinar el valor de complejidad máximo para el fragmento actual de los datos de vídeo comprenden:

medios para inicializar el valor de complejidad máximo a un valor máximo entre los valores de complejidad primero y segundo cuando el bloque anterior es un primer bloque en el fragmento actual y el bloque actual es un segundo bloque en el fragmento actual; y

5 medios para actualizar el valor de complejidad máximo a un valor máximo entre el valor de complejidad máximo y el valor de complejidad tercero.

15. Un medio de almacenamiento no transitorio legible por ordenador que tiene almacenadas en el mismo instrucciones que, cuando se ejecutan, hacen que un procesador de un dispositivo realice el procedimiento de cualquiera de las reivindicaciones 1 a 9.

10

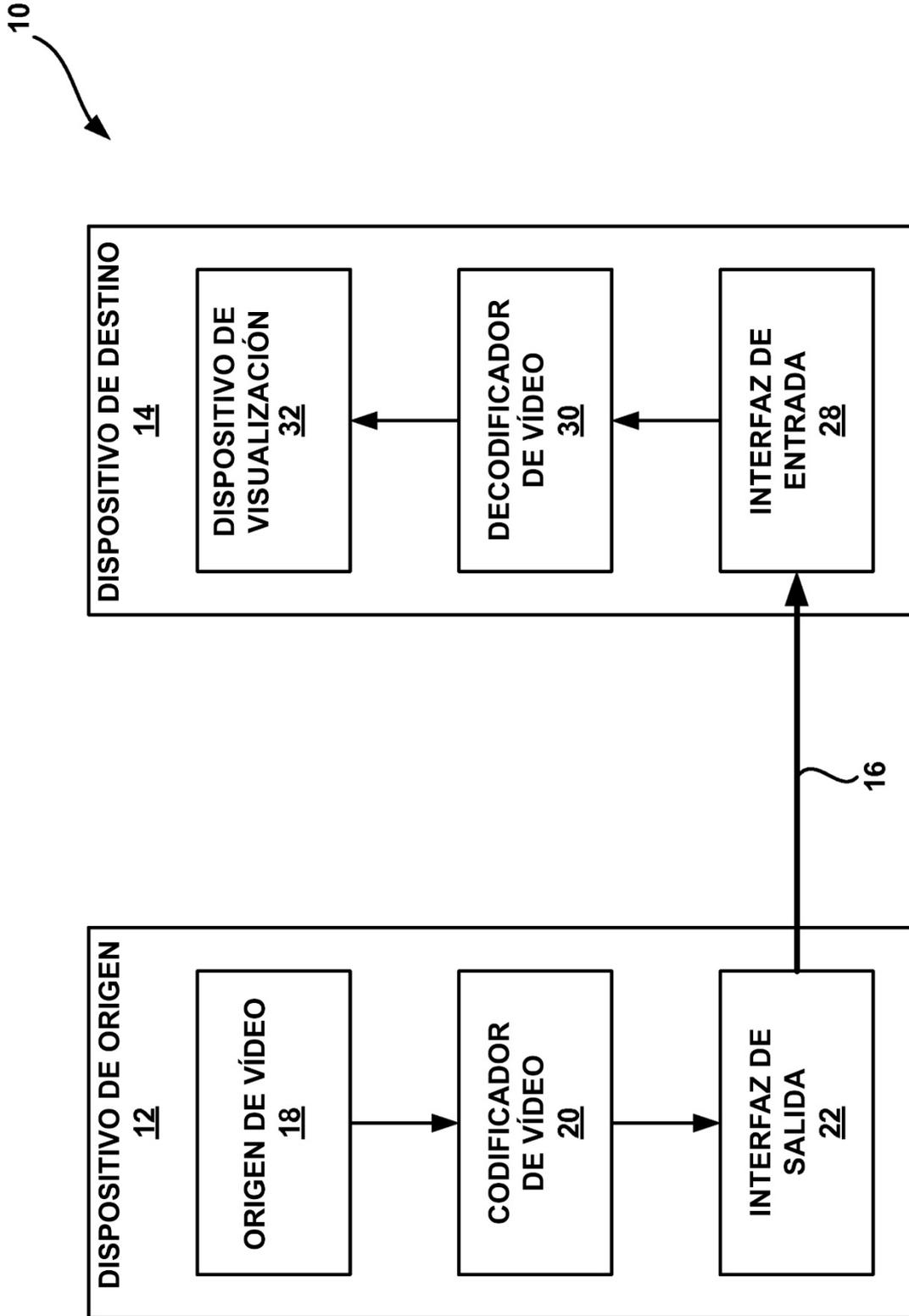


FIG. 1A

10'

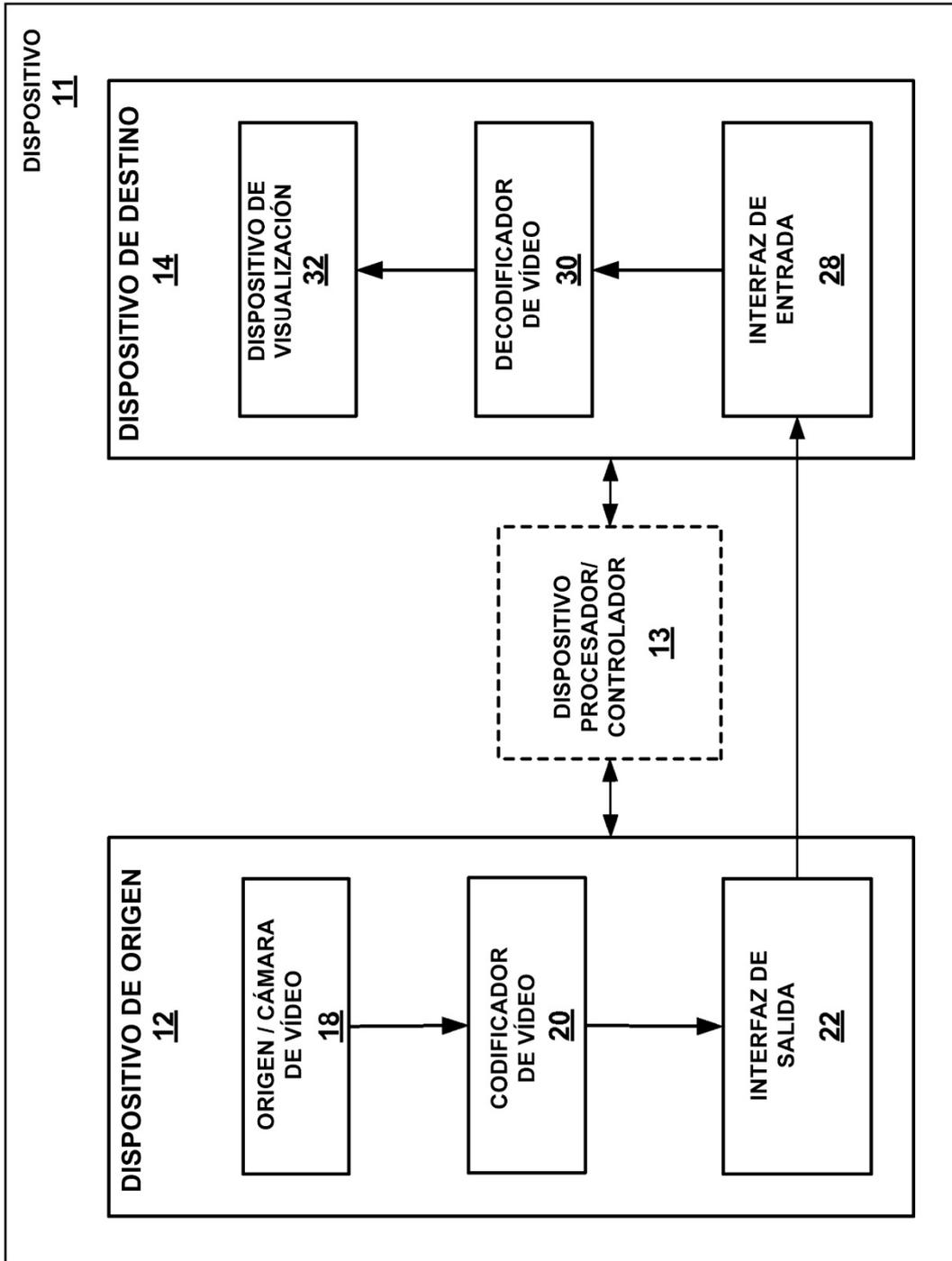


FIG. 1B

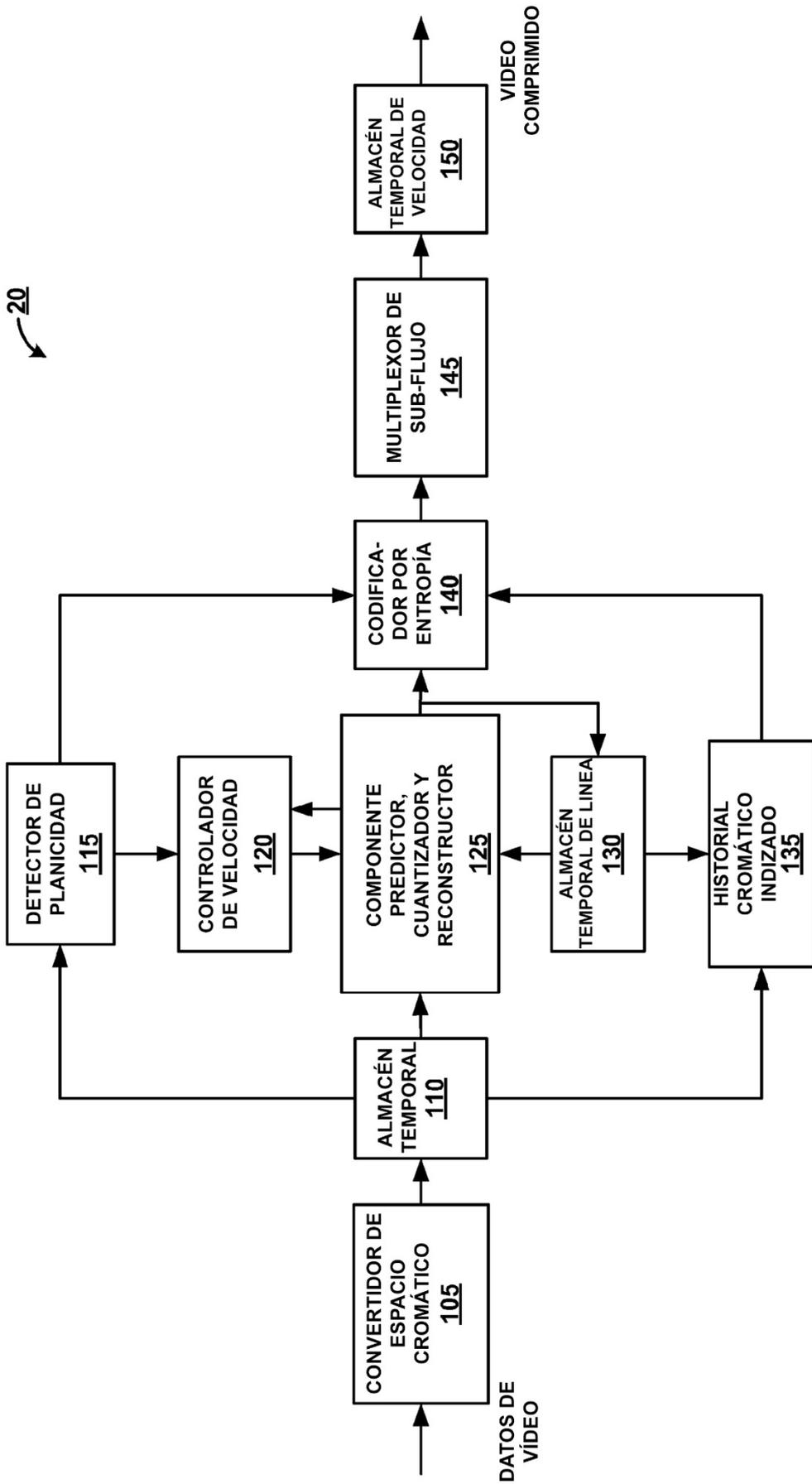


FIG. 2A

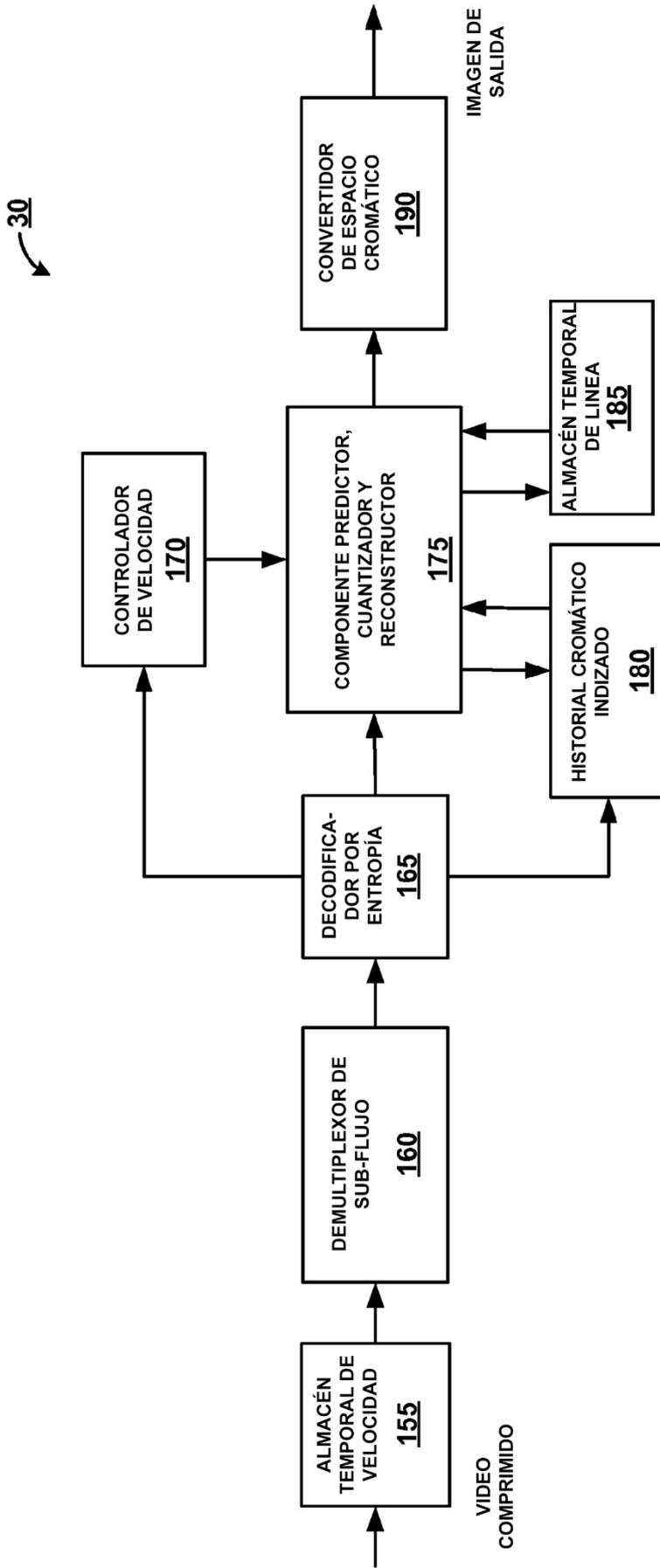


FIG. 2B

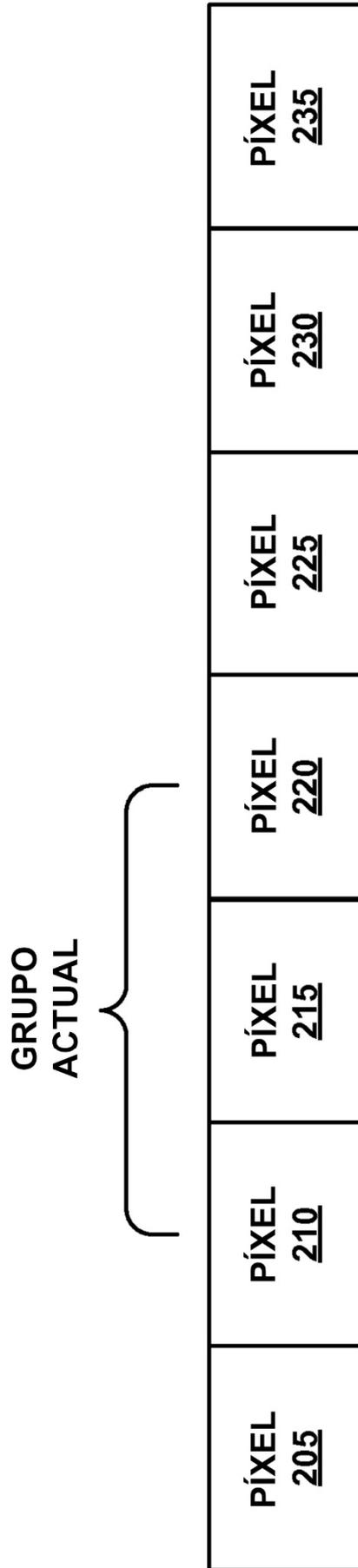


FIG. 3

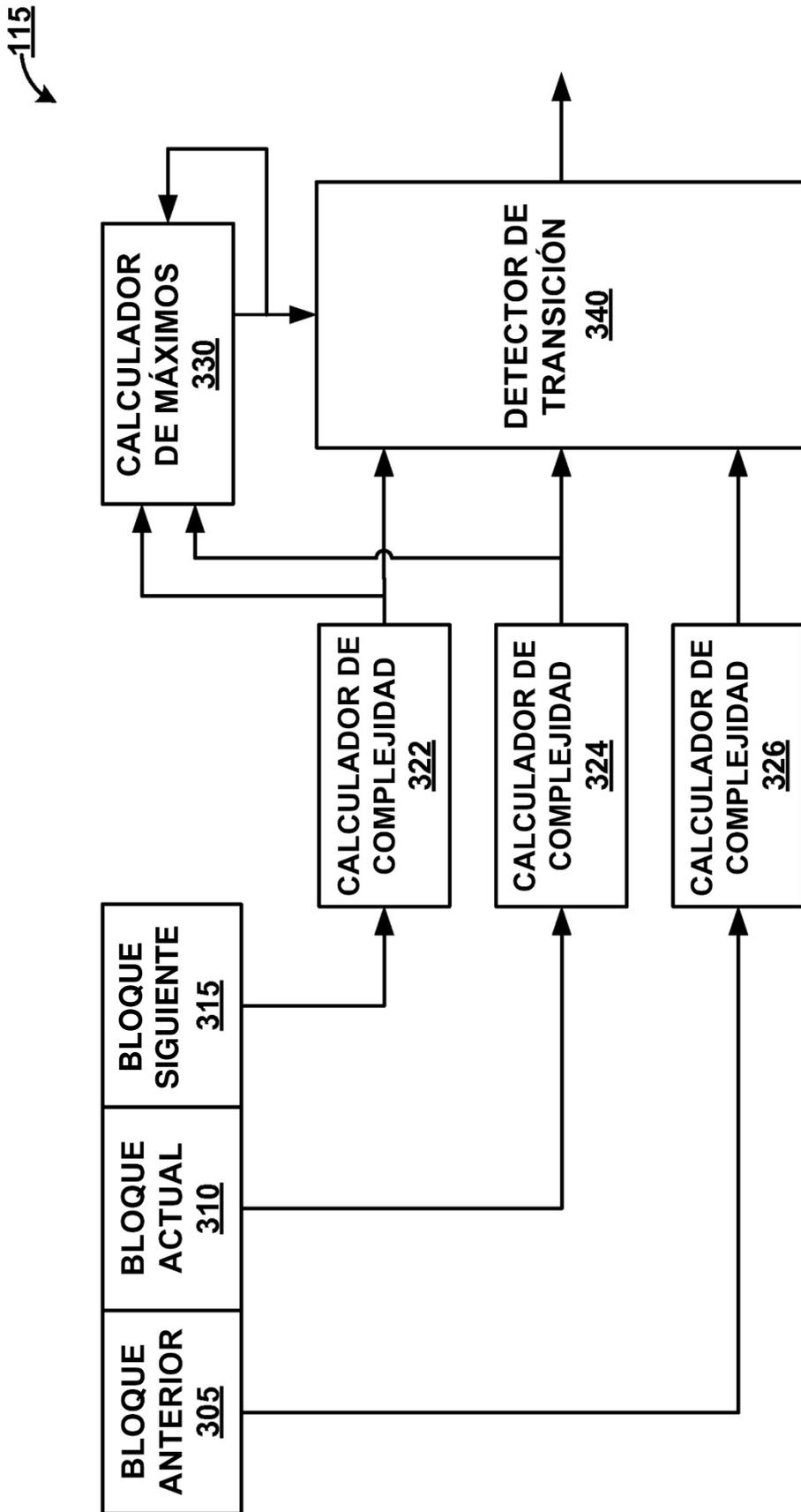


FIG. 4

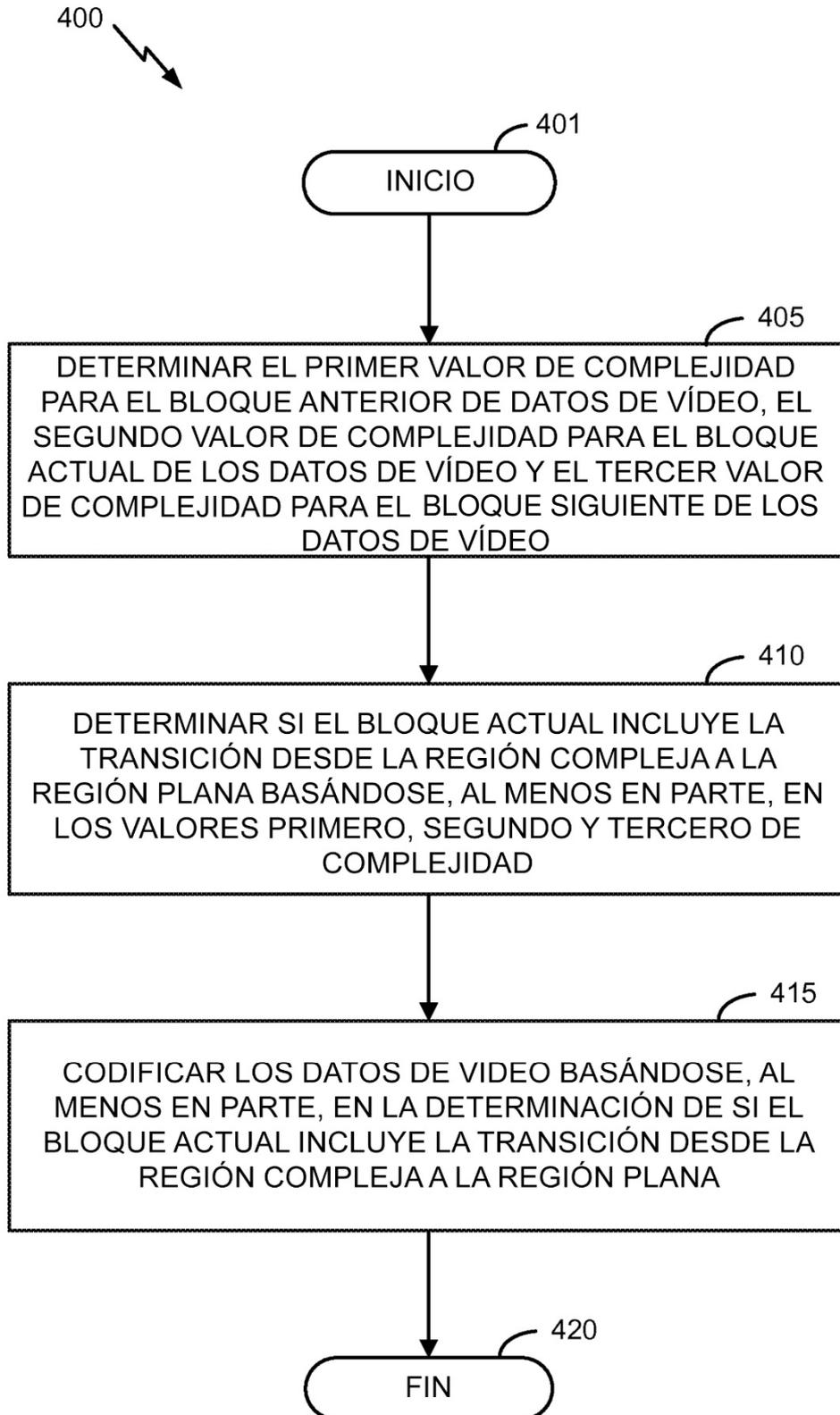


FIG. 5