

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 736 308**

51 Int. Cl.:

H04N 19/105 (2014.01)
H04N 19/154 (2014.01)
H04N 19/167 (2014.01)
H04N 19/182 (2014.01)
H04N 19/187 (2014.01)
H04N 19/30 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **20.03.2013 PCT/US2013/033068**

87 Fecha y número de publicación internacional: **26.09.2013 WO13142558**

96 Fecha de presentación y número de la solicitud europea: **20.03.2013 E 13713331 (0)**

97 Fecha y número de publicación de la concesión europea: **17.04.2019 EP 2829062**

54 Título: **Predicción de texturas entre capas para codificación de vídeo**

30 Prioridad:

22.03.2012 US 201261614450 P
28.09.2012 US 201261707620 P
15.03.2013 US 201313838270

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
27.12.2019

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121-1714, US

72 Inventor/es:

WANG, XIANGLIN y
KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 736 308 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

Predicción de texturas entre capas para codificación de vídeo

5 REFERENCIA CRUZADA CON SOLICITUDES RELACIONADAS

[0001] Esta solicitud reivindica prioridad a la solicitud provisional estadounidense n.º 61/614 450 presentada el 22 de marzo de 2012, la solicitud provisional estadounidense n.º 61/707 620 presentada el 28 de septiembre de 2012 y la solicitud de patente estadounidense n.º 13/838 270 presentada el 15 de marzo de 2013.

10

CAMPO TÉCNICO

[0002] Esta divulgación se refiere, en general, al campo de la codificación y compresión de vídeo y, más en particular, se refiere a la codificación de vídeo escalable (SVC) y a técnicas de predicción de texturas entre capas.

15

ANTECEDENTES

[0003] Las capacidades del vídeo digital pueden incorporarse a una amplia gama de dispositivos, incluidos televisores digitales, sistemas de difusión digital directa, sistemas de difusión inalámbrica, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, tablets electrónicas, lectores de libros electrónicos, cámaras digitales, dispositivos de grabación digital, reproductores de medios digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, los denominados "teléfonos inteligentes", dispositivos de videoconferencia, dispositivos de transmisión continua de vídeo y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en las normas definidas por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, parte 10, codificación de vídeo avanzada (AVC), la norma de codificación de vídeo de alta eficiencia (HEVC) actualmente en desarrollo y las ampliaciones de dichas normas. Los dispositivos de vídeo pueden transmitir, recibir, codificar, descodificar y/o almacenar información de vídeo digital con más eficacia implementando dichas técnicas de compresión de vídeo.

20

25

30

[0004] Las técnicas de compresión de vídeo realizan predicción espacial (intraimagen) y/o predicción temporal (entre imágenes) para reducir o eliminar la redundancia intrínseca a las secuencias de vídeo. Para la codificación de vídeo basada en bloques, un fragmento de vídeo (por ejemplo, una trama de vídeo o una parte de una trama de vídeo) puede dividirse en bloques de vídeo, que también pueden denominarse bloques de árbol, unidades de codificación (CU) y/o nodos de codificación. Los bloques de vídeo en un fragmento intracodificado (I) de una imagen se codifican usando predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen. Los bloques de vídeo en un fragmento intercodificado (P o B) de una imagen pueden usar la predicción espacial con respecto a muestras de referencia en bloques vecinos en la misma imagen o la predicción temporal con respecto a muestras de referencia en otras imágenes de referencia. Las imágenes pueden denominarse tramas y las imágenes de referencia pueden denominarse tramas de referencia.

35

40

[0005] La predicción espacial o temporal da como resultado un bloque predictivo para un bloque a codificar. Los datos residuales representan diferencias de píxeles entre el bloque original a codificar y el bloque predictivo. Un bloque intercodificado se codifica de acuerdo 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 del píxel a un dominio de transformación, dando como resultado coeficientes de transformación residuales, que a continuación se pueden cuantificar. Los coeficientes de transformación cuantificados, dispuestos inicialmente en una formación bidimensional, pueden explorarse con el fin de generar un vector unidimensional de coeficientes de transformación, y puede aplicarse codificación por entropía para lograr aún más compresión.

45

50

[0006] El documento "Quality Scalable Low Delay Video Coding Using Leaky Base Layer Prediction" [Codificación de vídeo de bajo retardo de calidad escalable mediante predicción de capa base permeable] de Steffen Kamp *et al.*, con fecha de 1 de octubre de 2007, analiza la codificación de vídeo de calidad escalable mediante predicción de capa base permeable para aplicaciones de bajo retardo. La referencia de predicción temporal para la capa base se genera calculando un promedio ponderado de las imágenes de referencia de la capa base de calidad y de la capa de mejora de la calidad. Esto proporciona un rendimiento aproximado a una sola capa en la cadencia de la capa de mejora al tiempo que introduce un desfase en la capa base si la capa de mejora se trunca o se pierde.

55

60

[0007] El documento "CE8: Verif JVT-U043 SVC-to-AVC" de Schwarz H, con fecha de 20 de octubre de 2006, analiza la inclusión de un modo especial para la codificación CGS en el diseño de SVC, que permite una reescritura directa de un flujo de bits SVC obteniéndose un flujo de bits AVC sin una transcodificación costosa.

65

[0008] El documento "Architectures for Fast Transcoding of H.264/AVC to Quality-Scalable SVC Streams" [Arquitecturas de transcodificación rápida de H.264/AVC para flujos SVC de calidad escalable] de Cock J *et al.*, con fecha de 1 de noviembre de 2009, analiza cómo se normalizó la ampliación escalable de H.264/AVC (SVC), y

ofrece escalabilidad con una penalización mínima en la eficiencia de distorsión de velocidad en comparación con la codificación H.264/AVC de una sola capa. En SVC, una versión escalada de la secuencia de vídeo original se puede extraer fácilmente eliminando capas del flujo.

5 **[0009]** El documento "Scalable Structures and Inter-Layer Predictions for HEVC scalable extension" [Estructuras escalables y predicciones entre capas para la ampliación escalable de HEVC] de Choi H M *et al.*, con fecha de 1 de julio de 2011, analiza una investigación en dos tipos de estructuras escalables y predicciones entre capas para una ampliación escalable de HEVC. Los dos tipos de estructuras son diseños de uno o varios bucles. En la contribución, se proponen dos predicciones entre capas diferentes que se pueden aplicar a dos estructuras escalables. Una predicción de textura entre capas (ILTP) es un esquema para predecir la PU de la capa de mejora a partir del bloque correspondiente de una capa de referencia. En la otra predicción entre capas, se presentará una trama de referencia generalizada entre capas (GILR). En esta contribución, cada estructura escalable se evalúa con dos predicciones entre capas.

15 **SUMARIO**

[0010] 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.

20 **[0011]** Un aspecto de esta divulgación proporciona un procedimiento para codificar vídeo digital, donde el procedimiento comprende obtener información de vídeo asociada con una capa base y una capa de mejora correspondiente, determinar un primer peso asociado con un predictor de capa base, Pb, y un segundo peso asociado con un predictor de capa de mejora, Pe, donde el predictor de capa base, Pb, se forma utilizando una textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente, y el predictor de capa de mejora, Pe, se forma usando la intrapredicción de un bloque de capa de mejora de un bloque vecino de píxeles reconstruidos, codificar un píxel actual en la trama de la capa de mejora en función de un predictor final, Pf, en el que el predictor final, Pf, comprende una combinación del predictor de capa base, Pb, ponderado de acuerdo con el primer peso y el predictor de capa de mejora, Pe, ponderado de acuerdo con el segundo peso, caracterizado por que el primer peso y el segundo peso se determinan en función de una distancia entre la ubicación de píxel codificado actual en un bloque de capa de mejora actual y un borde de un bloque de capa de mejora vecino, y en el que el segundo peso aumenta, en general, para los píxeles más cercanos al borde, el primer peso aumenta, en general, para los píxeles más alejados del borde, y para cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada.

35 **[0012]** Otro aspecto de la divulgación proporciona un procedimiento para descodificar vídeo digital, donde el procedimiento comprende obtener información de vídeo asociada con una capa base y una capa de mejora correspondiente, determinar un primer peso asociado con un predictor de capa base, Pb, y un segundo peso asociado con un predictor de capa de mejora, Pe, donde el predictor de capa base, Pb, se forma utilizando una textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente, y el predictor de capa de mejora, Pe, se forma usando la intrapredicción de un bloque de capa de mejora de un bloque vecino de píxeles reconstruidos, descodificar un píxel actual en la trama de la capa de mejora en función de un predictor final, Pf, en el que el predictor final, Pf, comprende una combinación del predictor de capa base, Pb, ponderado de acuerdo con el primer peso y el predictor de capa de mejora, Pe, ponderado de acuerdo con el segundo peso, caracterizado por que el primer peso y el segundo peso se determinan en función de una distancia entre la ubicación de píxel codificado actual en un bloque de capa de mejora actual y un borde de un bloque de capa de mejora vecino, y en el que el segundo peso aumenta, en general, para los píxeles más cercanos al borde, el primer peso aumenta, en general, para los píxeles más alejados del borde, y para cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada.

50 **[0013]** Otro aspecto adicional de la divulgación proporciona un aparato para codificar vídeo digital, donde el aparato comprende una memoria configurada para almacenar información de vídeo asociada con una capa base y una capa de mejora correspondiente; y un procesador en comunicación con la memoria, donde el procesador está configurado para determinar un primer peso asociado con un predictor de capa base, Pb, y un segundo peso asociado con un predictor de capa de mejora, Pe, donde el predictor de capa base, Pb, se forma utilizando una textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente, y el predictor de capa de mejora, Pe, se forma usando la intrapredicción de un bloque de capa de mejora de un bloque vecino de píxeles reconstruidos, y codificar un píxel actual en la trama de la capa de mejora en función de un predictor final, Pf, en el que el predictor final, Pf, comprende una combinación del predictor de capa base, Pb, ponderado de acuerdo con el primer peso y el predictor de capa de mejora, Pe, ponderado de acuerdo con el segundo peso, caracterizado por que el primer peso y el segundo peso se determinan en función de una distancia entre la ubicación de píxel codificado actual en un bloque de capa de mejora actual y un borde de un bloque de capa de mejora vecino, y en el que el segundo peso aumenta, en general, para los píxeles más cercanos al borde, el primer peso aumenta, en general, para los píxeles más alejados del borde, y para cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada.

[0014] Un aspecto adicional de la divulgación proporciona un aparato para descodificar vídeo digital, donde el aparato comprende una memoria configurada para almacenar información de vídeo asociada con una capa base y una capa de mejora correspondiente; y un procesador en comunicación con la memoria, donde el procesador está configurado para determinar un primer peso asociado con un predictor de capa base, P_b , y un segundo peso asociado con un predictor de capa de mejora, P_e , donde el predictor de capa base, P_b , se forma utilizando una textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente, y el predictor de capa de mejora, P_e , se forma usando la intrapredicción de un bloque de capa de mejora de un bloque vecino de píxeles reconstruidos, y descodificar un píxel actual en la trama de la capa de mejora en función de un predictor final, P_f , en el que el predictor final, P_f , comprende una combinación del predictor de capa base, P_b , ponderado de acuerdo con el primer peso y el predictor de capa de mejora, P_e , ponderado de acuerdo con el segundo peso, caracterizado por que el primer peso y el segundo peso se determinan en función de una distancia entre la ubicación de píxel codificado actual en un bloque de capa de mejora actual y un borde de un bloque de capa de mejora vecino, y en el que el segundo peso aumenta, en general, para los píxeles más cercanos al borde, el primer peso aumenta, en general, para los píxeles más alejados del borde, y para cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0015]

La FIG. 1A es un diagrama de bloques de un sistema de codificación y descodificación de vídeo ilustrativo que puede utilizar técnicas de acuerdo con los aspectos descritos en esta divulgación.

La FIG. 2 es un diagrama de flujo de un proceso ilustrativo para implementar las técnicas de codificación descritas en esta divulgación.

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

La FIG. 4 es un diagrama de bloques que ilustra un descodificador de vídeo de ejemplo que puede implementar las técnicas de codificación descritas en esta divulgación.

La FIG. 5 es un diagrama de modos de intrapredicción ilustrativos.

La FIG. 6 es una representación ilustrativa del proceso de predicción.

La FIG. 7 es un diagrama de un bloque de mejora actual ilustrativo a codificar, donde ciertos píxeles del bloque de mejora actual se identifican como estando relativamente cerca del borde de un bloque vecino.

Las FIG. 8A y 8B son diagramas de bloques de mejora actuales ilustrativos a codificar, donde ciertos píxeles de los bloques de mejora actuales se identifican como estando relativamente cerca del borde de un bloque vecino en función de una dirección de predicción.

La FIG. 9 es un diagrama de un bloque de mejora actual ilustrativo a codificar, donde los parámetros de ponderación varían en función de una dirección de predicción.

La FIG. 10 es un diagrama de flujo de un procedimiento ilustrativo para codificar vídeo usando predictores de capa base ponderados y predictores de capa de mejora.

La FIG. 11 es un diagrama de flujo de un procedimiento ilustrativo para ponderar y combinar predictores de capa base y predictores de capa de mejora en función de la ubicación de píxel.

La FIG. 12 es un diagrama de flujo de un procedimiento ilustrativo para ponderar y combinar predictores de capa base y predictores de capa de mejora en función de características particulares de la capa base y la capa de mejora.

DESCRIPCIÓN DETALLADA

[0016] En términos generales, esta divulgación se refiere a una codificación de vídeo escalable. Los aspectos de la divulgación se describirán con respecto al uso de predictores de capa base ponderados y predictores de capa de mejora en combinación para generar un predictor final para una capa de mejora. En algunos ejemplos, los pesos pueden ser constantes para un fragmento, un bloque de codificación o un flujo de bits completos. En ejemplos adicionales, los pesos para píxeles, fragmentos, bloques de codificación, etc. particulares pueden variar. Los factores que pueden considerarse al determinar pesos variables incluyen, entre otras cosas, la proximidad de un píxel a un bloque codificado o descodificado previamente y la dirección de intrapredicción. Otros factores incluyen parámetros de cuantificación y relaciones de escalamiento espacial.

5 **[0017]** En ejemplos adicionales, las condiciones sobre las cuales se determinan los pesos respectivos para los predictores de capa de mejora y los predictores de capa base pueden obtenerse implícitamente a partir de los predictores o de sus características. Por consiguiente, no es necesario transmitir datos adicionales a un descodificador; en cambio, el descodificador puede obtener dinámicamente la ponderación. En otros ejemplos adicionales, los datos pueden incluirse en el flujo de bits que el descodificador puede usar para determinar la ponderación.

10 **[0018]** Los detalles de uno o más ejemplos se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características, objetos y ventajas resultarán evidentes a partir de la descripción y de los dibujos, y a partir de las reivindicaciones.

15 **[0019]** Las imágenes digitales, tales como las imágenes de vídeo, imágenes de TV, imágenes fijas o imágenes generadas por una grabadora de vídeo o un ordenador, consisten en píxeles dispuestos en líneas horizontales y verticales. El número de píxeles en una sola imagen es típicamente de decenas de miles para imágenes más pequeñas y de millones para imágenes de alta definición. Cada píxel contiene típicamente información de luminancia y crominancia. Sin compresión, la cantidad de información que se transmitirá desde un codificador de imágenes a un descodificador de imágenes es tan grande que haría inviable la transmisión 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.

20 **[0020]** En un codificador de vídeo típico, la trama de la secuencia de vídeo original se puede dividir en regiones o bloques rectangulares, que se codifican en intramodo (modo I) o intermodo (modo P). Los bloques se pueden codificar utilizando algún tipo de codificación de transformación, como la codificación DCT. Sin embargo, la codificación basada en transformación pura solo reduce la correlación entre píxeles dentro de un bloque particular, sin tener en cuenta la correlación entre bloques de los píxeles y, en general, sigue produciendo altas velocidades de transmisión de bits para su transmisión. Las normas actuales de codificación de imágenes digitales también utilizan ciertos procedimientos que reducen la correlación de valores de píxeles entre bloques.

25 **[0021]** En general, los bloques codificados en modo P se predicen a partir de una de las tramas previamente codificadas y transmitidas. La información de predicción de un bloque se representa mediante un vector de movimiento bidimensional (2D). Para los bloques codificados en modo I, el bloque predicho puede formarse usando predicción espacial a partir de bloques vecinos previamente codificados dentro de la misma trama. El error de predicción, es decir, la diferencia entre el bloque que se está codificando y el bloque predicho se representa como un conjunto de funciones base ponderadas de alguna transformación discreta. La transformación se realiza típicamente en bloques de 8x8 o 4x4. Los pesos (por ejemplo, los coeficientes de transformación) se cuantifican posteriormente. La cuantificación introduce una pérdida de información y, por lo tanto, los coeficientes cuantificados pueden tener una precisión menor que los originales.

30 **[0022]** Los coeficientes de transformación cuantificados, junto con los vectores de movimiento y cierta información de control, forman una representación de secuencias codificadas completas y se denominan elementos de sintaxis. Antes de la transmisión desde el codificador al descodificador, los elementos de sintaxis pueden codificarse por entropía para reducir aún más el número de bits necesarios para su representación.

35 **[0023]** En el descodificador de vídeo, un bloque en una trama actual puede obtenerse construyendo primero su predicción de la misma manera que en el codificador, y añadiendo a la predicción el error de predicción comprimido. El error de predicción comprimido puede hallarse al ponderar las funciones base de transformación usando los coeficientes cuantificados. La diferencia entre la trama reconstruida y la trama original se denomina típicamente error de reconstrucción.

40 **[0024]** Las técnicas descritas en esta divulgación se refieren, en general, a la codificación de vídeo escalable (SVC) y la codificación de vídeo 3D. Por ejemplo, las técnicas pueden referirse a y utilizarse con o en una ampliación de codificación de vídeo escalable (SVC) de codificación de vídeo de alta eficiencia (HEVC). En una ampliación de SVC, podría haber múltiples capas de información de vídeo. La capa en el nivel más bajo puede servir como capa base (BL), y la capa en la parte superior puede servir como capa mejorada (EL). La "capa mejorada" a veces se denomina "capa de mejora", y estos términos se pueden usar indistintamente. Todas las capas centrales pueden servir como EL y/o BL. Por ejemplo, una capa central puede ser una EL para las capas por debajo de la misma, tal como la capa base o cualquier capa de mejora intermedia, y al mismo tiempo servir como RL para las capas de mejora dispuestas sobre la misma.

45 **[0025]** Para fines de ilustración solamente, las técnicas descritas en la divulgación se describen con ejemplos que incluyen solo dos capas (por ejemplo, una capa de nivel inferior, tal como la capa base, y una capa de nivel superior, tal como la capa mejorada). Debe entenderse que los ejemplos descritos en esta divulgación pueden extenderse también a ejemplos con múltiples capas base y capas de mejora.

50

[0026] Entre las normas de codificación de vídeo se incluyen ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 o ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (también conocida como ISO/IEC MPEG-4 AVC), incluidas sus ampliaciones de codificación de vídeo escalable (SVC) y de codificación de vídeo multivista (MVC). El último borrador conjunto de SVC y MVC se describe en "Advanced video coding for generic audiovisual services" [Codificación de vídeo avanzada para servicios audiovisuales genéricos], recomendación ITU-T H.264, marzo de 2010. Además, existe una nueva norma de codificación de vídeo, la codificación de vídeo de alta eficiencia (HEVC), que está siendo desarrollada por el Equipo de Colaboración Conjunta sobre Codificación de Vídeo (JCT-VC) del Grupo de Expertos sobre Codificación de Vídeo (VCEG) de ITU-T y el Grupo de Expertos sobre Imágenes en Movimiento (MPEG) de ISO/IEC. Un borrador de memoria descriptiva textual de codificación de vídeo de alta eficiencia (HEVC) reciente está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v13.zip. Otro borrador de trabajo (WD) reciente de HEVC y denominado HEVC WD9, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/11_Shanghai/wg11/JCTVC-K1003-v13.zip. Otro borrador de trabajo de HEVC, denominado HEVC WD8 (o WD8) está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/10_Stockholm/wg11/JCTVC-J1003-v8.zip. Otro borrador de trabajo de HEVC y denominado HEVC WD7 en adelante en el presente documento, está disponible en http://phenix.int-evry.fr/jct/doc_end_user/documents/9_Geneva/wg11/JCTVC-I1003-v5.zip.

[0027] En la codificación de vídeo escalable (SVC), los flujos de bits de cada capa (por ejemplo, una capa base y una capa de mejora) se multiplexan juntos en un solo flujo de bits. Tal flujo de bits es escalable en el sentido de que los flujos de bits de capa de mejora, cuando se descodifican, pueden proporcionar ciertas mejoras a la capa base. Tales mejoras incluyen resolución espacial, resolución temporal, mejoras de calidad, y similares, en relación con la capa base. La capa base puede descodificarse independientemente de las capas de mejora. De manera correspondiente, dichas mejoras también se denominan respectivamente escalabilidad espacial, escalabilidad temporal y escalabilidad de relación de señal a ruido (SNR).

[0028] Independientemente del tipo de escalabilidad, el objetivo de SVC es, en general, utilizar la correlación entre capas para mejorar la eficiencia de la codificación. Dicha correlación entre capas puede existir en diferentes sintaxis de bloques correspondientes en diferentes capas (por ejemplo, modos de predicción, vectores de movimiento, residuos de predicción, etc.).

[0029] Para utilizar tales correlaciones, en el pasado se han propuesto varias herramientas de codificación. En la ampliación escalable de H.264/AVC, por ejemplo, se definen varias herramientas de codificación, tal como el modo de capa intrabase (BL), la predicción residual, la herencia de modo y la predicción de vector de movimiento. En el intramodo de BL, la textura de un bloque reconstruido de capa base se utiliza como predictor para predecir el bloque de capa de mejora correspondiente. Con la predicción residual, el residuo de predicción de un bloque de capa base se usa para predecir el residuo de predicción de un bloque de capa de mejora correspondiente. Con la herencia de modo, el modo de predicción (incluido el modo de partición) de un bloque de capa base se utiliza para predecir el modo de predicción de un bloque de capa de mejora. Con la predicción de vector de movimiento, los vectores de movimiento de un bloque de capa base se utilizan para predecir los vectores de movimiento de un bloque de capa de mejora.

[0030] En SVC, el que una capa sea una capa base o una capa de mejora puede considerarse, en general, como algo relativo. Por ejemplo, cualquier capa entre una primera capa y una última capa puede ser una capa de mejora para una o más capas inferiores y, al mismo tiempo, puede servir como una capa base para una o más capas superiores.

[0031] SVC se puede usar para proporcionar escalabilidad de calidad (también llamada relación de señal a ruido (SNR)), escalabilidad espacial y/o escalabilidad temporal. Una capa mejorada puede tener una resolución espacial diferente a una capa base. Por ejemplo, la relación de aspecto espacial entre la EL y la BL puede ser 1,0, 1,5, 2,0 u otras relaciones diferentes. En otras palabras, el aspecto espacial de la EL puede ser igual a 1,0, 1,5 o 2,0 veces el aspecto espacial de la BL. En algunos ejemplos, el factor de escalamiento de la EL puede ser mayor que la del BL. Por ejemplo, un tamaño de imágenes en la EL puede ser mayor que un tamaño de imágenes en la BL. De esta manera, puede ser posible, aunque no es una limitación, que la resolución espacial de la EL sea mayor que la resolución espacial de la BL.

[0032] Los aspectos de la presente divulgación se refieren al uso de predictores combinados, basados en predictores tanto de la EL como de la BL, para generar un bloque de EL predicho. Al codificar una EL se pueden generar predictores como se ha descrito en general anteriormente. Después, los predictores de EL se pueden combinar con predictores de BL para mejorar la calidad del predictor final que se utiliza para la predicción entre capas de un píxel de EL. Al combinar los predictores de EL y los predictores de BL, la ponderación se puede aplicar de manera que el predictor de EL afecte el predictor final más que el predictor de BL en casos en los que es probable que el predictor de EL sea más preciso. De forma alternativa, la ponderación se puede aplicar de manera que el predictor de BL afecte el predictor final más que el predictor de EL en casos en los que es probable que el predictor de BL sea más preciso. En casos en los que se desconoce cuál de los predictores será el más preciso, el predictor de BL y el predictor de EL pueden ponderarse de manera equitativa.

[0033] En lo sucesivo se describen de forma más detallada diversos aspectos de los sistemas, aparatos y procedimientos novedosos, con referencia a los dibujos adjuntos. Sin embargo, esta divulgación se puede realizar de muchas formas diferentes y no se debería interpretar que está limitada a ninguna 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 materia. Basándose en las enseñanzas del presente documento, un experto en la técnica debería apreciar que el alcance de la divulgación está concebido para abarcar cualquier aspecto de los sistemas, aparatos y procedimientos novedosos divulgados en el presente documento, ya sea implementados de forma independiente de, o en combinación con, cualquier otro aspecto de la invención. 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 invención está concebido para abarcar uno de dichos aparatos o procedimientos, que se lleva a la práctica usando otra estructura, funcionalidad o estructura y funcionalidad, de forma adicional o alternativa a los diversos aspectos de la invención expuestos en el presente documento. Debería entenderse que cualquier aspecto divulgado en el presente documento puede realizarse mediante uno o más elementos de una reivindicación.

[0034] 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 preferentes, el alcance de la divulgación no pretende 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 preferentes. La descripción detallada y los dibujos simplemente ilustran la divulgación y no limitan el alcance de la divulgación, la cual está definida por las reivindicaciones adjuntas.

[0035] La FIG. 1A es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo de ejemplo que puede utilizar técnicas de acuerdo con los aspectos descritos en esta divulgación. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que proporciona datos de vídeo codificados que se descodificarán posteriormente por un dispositivo de destino 14. En particular, el dispositivo de origen 12 proporciona los datos de vídeo al dispositivo de destino 14 a través de un medio legible por ordenador 16. El dispositivo de origen 12 y el dispositivo de destino 14 pueden comprender cualquiera de una amplia gama de dispositivos, incluidos ordenadores de escritorio, ordenadores portátiles, tablets electrónicas, descodificadores, equipos telefónicos de mano tales como los denominados teléfonos "inteligentes", los denominados paneles "inteligentes", televisores, cámaras, dispositivos de visualización, reproductores de medios digitales, consolas de videojuegos, dispositivos de transmisión continua de vídeo o similares. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 14 pueden estar equipados para la comunicación inalámbrica.

[0036] El dispositivo de destino 14 puede recibir los datos de vídeo codificados a descodificar mediante el medio legible por ordenador 16. El medio legible por ordenador 16 puede comprender cualquier tipo de medio o dispositivo capaz de transportar los datos de vídeo codificados desde el dispositivo de origen 12 al dispositivo de destino 14. En un ejemplo, el medio legible por ordenador 16 puede comprender un medio de comunicación para habilitar el dispositivo de origen 12 para transmitir datos de vídeo codificados directamente al dispositivo de destino 14 en tiempo real. Los datos de vídeo codificados se pueden modular de acuerdo con una norma de comunicación, tal como un protocolo de comunicación inalámbrica, y transmitir al dispositivo de destino 14. El medio de comunicación puede comprender cualquier medio de comunicación, inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas físicas de transmisión. 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 hasta el dispositivo de destino 14.

[0037] En algunos ejemplos, pueden emitirse datos codificados desde la interfaz de salida 22 hasta un dispositivo de almacenamiento. De forma similar, se puede acceder a los datos codificados desde el dispositivo de almacenamiento mediante una interfaz de entrada. El dispositivo de almacenamiento puede incluir cualquiera de varios medios de almacenamiento de datos, distribuidos o de acceso local, tales como un disco duro, discos Blu-ray, DVD, CD-ROM, memoria flash, memoria volátil o no volátil, o cualquier otro medio de almacenamiento digital adecuado para almacenar datos de vídeo codificados. En un ejemplo adicional, el dispositivo de almacenamiento puede corresponder a un servidor de archivos o a otro dispositivo de almacenamiento intermedio que pueda almacenar el vídeo codificado generado por el dispositivo de origen 12. El dispositivo de destino 14 puede acceder a datos de vídeo almacenados desde el dispositivo de almacenamiento, mediante transmisión continua o descarga. El servidor de archivos puede ser cualquier tipo de servidor capaz de almacenar datos de vídeo codificados y transmitir esos datos de vídeo codificados al dispositivo de destino 14. Los servidores de archivos a modo de ejemplo incluyen un servidor de red (por ejemplo, para un sitio web), un servidor de FTP, dispositivos de almacenamiento conectados en red (NAS) o una unidad de disco local. El dispositivo de destino 14 puede acceder a datos de vídeo codificados a través de cualquier conexión de datos estándar, incluida una conexión a Internet.

Esto puede incluir un canal inalámbrico (por ejemplo, una conexión de Wi-Fi), una conexión cableada (por ejemplo, DSL, módem de cable, etc.) o una combinación de ambas que sea adecuada para acceder a datos de vídeo codificados almacenados en un servidor de archivos. La transmisión de datos de vídeo codificados desde el dispositivo de almacenamiento puede ser una transmisión continua, una transmisión de descarga o una combinación de ambas.

[0038] Las técnicas de esta divulgación no están limitadas necesariamente a aplicaciones o configuraciones inalámbricas. Las técnicas se pueden aplicar a la codificación de vídeo como soporte de cualquiera de varias aplicaciones multimedia, tales como difusiones de televisión por el aire, transmisiones de televisión por cable, transmisiones de televisión por satélite, transmisiones continuas de vídeo por Internet, tales como la transmisión continua adaptativa dinámica por HTTP (DASH), vídeo digital que se codifica en un medio de almacenamiento de datos, descodificación de vídeo digital almacenado en un medio de almacenamiento de datos u otras aplicaciones. En algunos ejemplos, el sistema 10 puede estar configurado para prestar soporte a una transmisión de vídeo unidireccional o bidireccional, a fin de prestar soporte a aplicaciones tales como la transmisión continua de vídeo, la reproducción de vídeo, la difusión de vídeo y/o la videotelefonía.

[0039] 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. El dispositivo de destino 14 incluye una interfaz de entrada 28, un decodificador de vídeo 30 y un dispositivo de visualización 32. De acuerdo con esta divulgación, el codificador de vídeo 20 del dispositivo de origen 12 puede estar configurado para aplicar las técnicas para codificar un flujo de bits que incluye datos de vídeo según múltiples normas o ampliaciones de normas. En otros ejemplos, un dispositivo de origen y un dispositivo de destino pueden incluir otros componentes o disposiciones. Por ejemplo, el dispositivo de origen 12 puede recibir datos de vídeo desde una fuente de vídeo externa 18, tal como una cámara externa. Del mismo modo, el dispositivo de destino 14 puede interactuar con un dispositivo de visualización externo, en lugar de incluir un dispositivo de visualización integrado.

[0040] El sistema ilustrado 10 de la FIG. 1 es simplemente un ejemplo. Cualquier dispositivo de codificación y/o descodificación de vídeo digital puede realizar técnicas para determinar candidatos para una lista de candidatos para predictores de vectores de movimiento para un bloque actual. Aunque, en general, las técnicas de esta divulgación son llevadas a cabo por un dispositivo de codificación de vídeo, las técnicas también pueden ser llevadas a cabo por un codificador/descodificador de vídeo, denominado típicamente "CÓDEC". Además, las técnicas de esta divulgación también se pueden llevar a cabo mediante un preprocesador de vídeo. El dispositivo de origen 12 y el dispositivo de destino 14 son simplemente ejemplos de dichos dispositivos de codificación en los que el dispositivo de origen 12 genera datos de vídeo codificados para su transmisión al dispositivo de destino 14. En algunos ejemplos, los dispositivos 12, 14 pueden funcionar de manera esencialmente simétrica, de modo que cada uno de los dispositivos 12, 14 incluye componentes de codificación y descodificación de vídeo. De ahí que el sistema 10 pueda prestar soporte a la transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 14, por ejemplo, para la transmisión continua de vídeo, la reproducción de vídeo, la difusión de vídeo o la videotelefonía.

[0041] La fuente de vídeo 18 del dispositivo de origen 12 puede incluir un dispositivo de captura de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo capturado previamente y/o una interfaz de suministro de vídeo para recibir vídeo desde un proveedor de contenidos de vídeo. Como otra alternativa, la fuente de vídeo 18 puede generar datos basados en gráficos de ordenador como el vídeo de origen, o una combinación de vídeo en directo, vídeo archivado y vídeo generado por ordenador. En algunos casos, si la fuente de vídeo 18 es una videocámara, el dispositivo de origen 12 y el dispositivo de destino 14 pueden formar los llamados teléfonos con cámara o videoteléfonos. Sin embargo, como se ha mencionado anteriormente, las técnicas descritas en esta divulgación pueden ser aplicables a la codificación de vídeo en general, y se pueden aplicar a aplicaciones inalámbricas y/o cableadas. En cada caso, el codificador de vídeo 20 puede codificar el vídeo capturado, precapturado o generado por ordenador. Después, la información de vídeo codificada se puede proporcionar por la interfaz de salida 22 a un medio legible por ordenador 16.

[0042] El medio legible por ordenador 16 puede incluir medios transitorios, tales como una difusión inalámbrica o una transmisión de red cableada, o medios de almacenamiento (es decir, medios de almacenamiento no transitorios), tales como un disco duro, una unidad de memoria flash, un disco compacto, un disco de vídeo digital, un disco Blu-ray u otros medios legibles por ordenador. En algunos ejemplos, un servidor de red (no mostrado) puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y proporcionar los datos de vídeo codificados al dispositivo de destino 14, por ejemplo, a través de una transmisión en red, comunicación por cable directa, etc. De manera similar, un dispositivo informático de una instalación de producción de medios, tal como una instalación de estampado de discos, puede recibir datos de vídeo codificados desde el dispositivo de origen 12 y producir un disco que contiene los datos de vídeo codificados. Por lo tanto, se puede entender que el medio legible por ordenador 16 incluye uno o más medios legibles por ordenador de diversas formas, en varios ejemplos.

[0043] La interfaz de entrada 28 del dispositivo de destino 14 recibe información desde el medio legible por ordenador 16. La información del medio legible por ordenador 16 puede incluir información sintáctica definida por el codificador de vídeo 20, también usada por el decodificador de vídeo 30, que incluye elementos sintácticos que

describen características y/o procesamiento de bloques y otras unidades codificadas, por ejemplo, los GOP. El dispositivo de visualización 32 muestra los datos de vídeo descodificados a un usuario, y puede comprender cualquiera de varios dispositivos de visualización, tales como un tubo de rayos catódicos (CRT), una pantalla de cristal líquido (LCD), una pantalla de plasma, una pantalla de diodos orgánicos de emisión de luz (OLED) u otro tipo de dispositivo de visualización.

[0044] El codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con una norma de codificación de vídeo, tal como la norma de codificación de vídeo de alta eficacia (HEVC), actualmente en fase de desarrollo, y pueden ajustarse al modelo de prueba de HEVC (HM). De forma alternativa, el codificador de vídeo 20 y el descodificador de vídeo 30 pueden funcionar de acuerdo con otras normas de propiedad o industriales, tales como la norma ITU-T H.264, denominada de forma alternativa MPEG-4, parte 10, codificación avanzada de vídeo (AVC), o ampliaciones de dichas normas. Sin embargo, las técnicas de esta divulgación no se limitan a ninguna norma de codificación particular, incluidas, pero sin limitarse a, cualquiera de las normas enumeradas anteriormente. Otros ejemplos de normas de codificación de vídeo incluyen MPEG-2 e ITU-T H.263. Aunque no se muestra en la FIG. 1, en algunos aspectos, tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden estar integrados con un codificador y descodificador de audio, y pueden incluir unidades MUX-DEMUX adecuadas, u otro hardware y software, para ocuparse de la codificación tanto de audio como de vídeo en un flujo de datos común o en flujos de datos independientes. Si procede, las unidades MUX-DEMUX se pueden ajustar al protocolo multiplexador ITU H.223 o a otros protocolos tales como el protocolo de datagramas de usuario (UDP).

[0045] Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 pueden implementarse como cualquiera de varios sistemas 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 usando uno o más procesadores para realizar las técnicas de esta divulgación. Tanto el codificador de vídeo 20 como el descodificador de vídeo 30 se puede incluir en uno o más codificadores o descodificadores, cualquiera de los cuales se puede integrar como parte de un codificador/descodificador (CÓDEC) combinado en un dispositivo respectivo. Un dispositivo que incluye el codificador de vídeo 20 y/o el descodificador de vídeo 30 puede comprender un circuito integrado, un microprocesador y/o un dispositivo de comunicación inalámbrica, tal como un teléfono celular.

[0046] El equipo JCT-VC está trabajando en la elaboración de la norma HEVC. La iniciativa de normalización HEVC se basa en un modelo en evolución de un dispositivo de codificación de vídeo denominado modelo de prueba HEVC (HM). El HM supone varias capacidades adicionales de los dispositivos de codificación de vídeo respecto a los dispositivos existentes de acuerdo con, por ejemplo, la norma ITU-T H.264/AVC. Por ejemplo, mientras que la norma H.264 proporciona nueve modos de codificación mediante intrapredicción, el HM puede proporcionar hasta treinta y tres modos de codificación mediante intrapredicción.

[0047] En general, el modelo de trabajo del HM especifica que una trama o imagen de vídeo puede dividirse en una secuencia de bloques de árbol o unidades de codificación más grandes (LCU), que incluyen muestras tanto de luminancia como de crominancia. Los datos sintácticos dentro de un flujo de bits pueden definir un tamaño para la LCU, que es la mayor unidad de codificación en lo que respecta al número de píxeles. Un fragmento incluye una pluralidad de bloques de árbol consecutivos en orden de codificación. Una trama de vídeo o imagen puede dividirse en uno o más fragmentos. Cada bloque de árbol puede dividirse en unidades de codificación (CU) de acuerdo con un árbol cuaternario. En general, una estructura de datos de árbol cuaternario incluye un nodo por CU, con un nodo raíz correspondiente al bloque de árbol. Si una CU se divide en cuatro sub-CU, el nodo correspondiente a la CU incluye cuatro nodos hoja, cada uno de los cuales corresponde a una de las sub-CU.

[0048] Cada nodo de la estructura de datos en árbol cuaternario puede proporcionar datos sintácticos para la CU correspondiente. Por ejemplo, un nodo en el árbol cuaternario puede incluir un indicador de división, que indica si la CU correspondiente al nodo está dividida en varias sub-CU. Los elementos sintácticos para una CU pueden definirse de manera recursiva y pueden depender de si la CU está dividida en varias sub-CU. Si una CU no está dividida adicionalmente, se denomina CU hoja. En esta divulgación, cuatro sub-CU de una CU hoja también se denominarán CU hoja incluso aunque no haya ninguna división explícita de la CU hoja original. Por ejemplo, si una CU con un tamaño de 16x16 no se divide adicionalmente, las cuatro sub-CU de tamaño 8x8 también se denominarán CU hoja aunque la CU de tamaño 16x16 no se haya dividido nunca.

[0049] Una CU tiene un propósito similar a un macrobloque de la norma H.264, excepto porque una CU no tiene una distinción de tamaño. Por ejemplo, un bloque de árbol puede dividirse en cuatro nodos hijo (también denominados sub-CU) y cada nodo hijo puede a su vez ser un nodo padre y dividirse en otros cuatro nodos hijo. Un nodo hijo final, no dividido, denominado nodo hoja del árbol cuaternario, comprende un nodo de codificación, también denominado CU hoja. Los datos sintácticos asociados a un flujo de bits codificado pueden definir un número máximo de veces en que puede dividirse un bloque de árbol, lo que se denomina profundidad de CU máxima, y también pueden definir un tamaño mínimo de los nodos de codificación. Por consiguiente, un flujo de

bits también puede definir la unidad de codificación más pequeña (SCU). Esta divulgación utiliza el término "bloque" para referirse a cualquiera de una CU, PU o TU, en el contexto de HEVC, o a estructuras de datos similares en el contexto de otras normas (por ejemplo, macrobloques y subbloques de los mismos en la norma H.264/AVC).

5 **[0050]** Una CU incluye un nodo de codificación y unidades de predicción (PU) y unidades de transformación (TU) asociadas al nodo de codificación. Un tamaño de la CU corresponde a un tamaño del nodo de codificación y debe ser de forma cuadrada. El tamaño de la CU puede variar desde 8x8 píxeles hasta el tamaño del bloque de árbol, con un máximo de 64x64 píxeles o más. Cada CU puede contener una o más PU y una o más TU. Los datos sintácticos asociados a una CU pueden describir, por ejemplo, la división de la CU en una o más PU. Los modos de división pueden diferir entre si la CU está codificada en modo de salto o modo directo, codificada en modo de intrapredicción o codificada en modo de interpredicción. Las PU pueden dividirse para tener forma no cuadrada. Los datos sintácticos asociados a una CU también pueden describir, por ejemplo, la división de la CU en una o más TU de acuerdo con un árbol cuaternario. Una TU puede tener forma cuadrada o no cuadrada (por ejemplo, rectangular).

15 **[0051]** La norma HEVC admite transformaciones de acuerdo con las TU, que pueden ser diferentes para diferentes CU. El tamaño de las TU se basa típicamente en el tamaño de las PU dentro de una CU dada definida para una LCU dividida, aunque puede que no sea siempre así. Las TU son típicamente del mismo tamaño, o de un tamaño más pequeño, que las PU. En algunos ejemplos, las muestras residuales correspondientes a una CU pueden subdividirse en unidades más pequeñas utilizando una estructura de árbol cuaternario conocida como "árbol cuaternario residual" (RQT). Los nodos hoja del RQT pueden denominarse unidades de transformación (TU). Los valores de diferencia de píxeles asociados a las TU pueden transformarse para generar coeficientes de transformación, que pueden cuantificarse.

25 **[0052]** Una CU hoja puede incluir una o más unidades de predicción (PU). En general, una PU representa una zona espacial correspondiente a la totalidad, o a una parte, de la CU correspondiente, y puede incluir datos para recuperar una muestra de referencia para la PU. Además, una PU incluye datos relacionados con la predicción. Por ejemplo, cuando la PU está codificada mediante intramodo pueden incluirse datos para la PU en un árbol cuaternario residual (RQT), que puede incluir datos que describen un modo de intrapredicción para una TU correspondiente a la PU. Como otro ejemplo, cuando la PU está codificada mediante intermodo, la PU puede incluir datos que definen uno o más vectores de movimiento para la PU. Los datos que definen el vector de movimiento para una PU pueden describir, por ejemplo, una componente horizontal del vector de movimiento, una componente vertical del vector de movimiento, una resolución para el vector de movimiento (por ejemplo, precisión de un cuarto de píxel o precisión de un octavo de píxel), una imagen de referencia a la que apunta el vector de movimiento y/o una lista de imágenes de referencia (por ejemplo, lista 0, lista 1 o lista C) para el vector de movimiento.

35 **[0053]** Una CU hoja que tiene una o más PU también puede incluir una o más unidades de transformación (TU). Las unidades de transformación pueden especificarse usando un RQT (también denominado estructura de árbol cuaternario de TU), como se ha expuesto anteriormente. Por ejemplo, un indicador de división puede indicar si una CU hoja está dividida en cuatro unidades de transformación. A continuación, cada unidad de transformación puede dividirse adicionalmente en sub-TU adicionales. Cuando una TU no se divide adicionalmente, puede denominarse una TU hoja. En general, en lo que respecta a la intracodificación, todas las TU hoja que pertenecen a una CU hoja comparten el mismo modo de intrapredicción. Es decir, el mismo modo de intrapredicción se aplica en general para calcular valores predichos para todas las TU de una CU hoja. En lo que respecta a la intracodificación, un codificador de vídeo puede calcular un valor residual para cada TU hoja usando el modo de intrapredicción, como una diferencia entre la parte de la CU correspondiente a la TU y el bloque original. Una TU no está necesariamente limitada al tamaño de una PU. De este modo, las TU pueden ser mayores o menores que una PU. En lo que respecta a la intracodificación, una PU puede estar coubicada con una TU hoja correspondiente para la misma CU. En algunos ejemplos, el tamaño máximo de una TU hoja puede corresponder al tamaño de la CU hoja correspondiente.

40 **[0054]** Además, las TU de las CU hoja también pueden asociarse a estructuras de datos de árbol cuaternario respectivas, denominadas árboles cuaternarios residuales (RQT). Es decir, una CU hoja puede incluir un árbol cuaternario que indica cómo la CU hoja está dividida en varias TU. El nodo raíz de un árbol cuaternario de TU corresponde en general a una CU hoja, mientras que el nodo raíz de un árbol cuaternario de CU corresponde en general a un bloque de árbol (o LCU). Las TU del RQT que no están divididas se denominan TU hoja. En general, esta divulgación usa los términos CU y TU para hacer referencia a una CU hoja y a una TU hoja, respectivamente, a no ser que se indique lo contrario.

50 **[0055]** Una secuencia de vídeo incluye típicamente una serie de tramas de vídeo o imágenes. Un grupo de imágenes (GOP) comprende, en general, una serie de una o más de las imágenes de vídeo. Un GOP puede incluir datos sintácticos en una cabecera del GOP, una cabecera de una o más de las imágenes, o de otras ubicaciones, que describen una serie de imágenes incluidas en el GOP. Cada fragmento de una imagen puede incluir datos sintácticos de fragmento que describen un modo de codificación para el fragmento respectivo. El codificador de vídeo 20 actúa típicamente en bloques de vídeo dentro de fragmentos de vídeo individuales con el fin de codificar los datos de vídeo. Un bloque de vídeo puede corresponder a un nodo de codificación dentro de una CU. Los

bloques de vídeo pueden tener tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada.

[0056] Como un ejemplo, el HM admite predicción en diversos tamaños de PU. Si se supone que el tamaño de una CU particular es $2N \times 2N$, el HM admite intrapredicción en tamaños de PU de $2N \times 2N$ o $N \times N$ e interpredicción en tamaños de PU simétricos de $2N \times 2N$, $2N \times N$, $N \times 2N$ o $N \times N$. El HM también admite la división asimétrica para interpredicción en tamaños de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ y $nR \times 2N$. En la división asimétrica, una dirección de una CU no está dividida, mientras que la otra dirección está dividida en un 25 % y un 75 %. La parte de la CU correspondiente a la partición del 25 % está indicada por una "n" seguida de una indicación de "arriba", "abajo", "izquierda" o "derecha". Así pues, por ejemplo, " $2N \times nU$ " se refiere a una CU de tamaño $2N \times 2N$ que está dividida horizontalmente, con una PU de tamaño $2N \times 0,5N$ encima y una PU de tamaño $2N \times 1,5N$ debajo.

[0057] En esta divulgación, " $N \times N$ " y "N por N" pueden usarse indistintamente para hacer referencia a las dimensiones de píxeles de un bloque de vídeo en términos de dimensiones verticales y horizontales, por ejemplo, 16×16 píxeles o 16 por 16 píxeles. En general, un bloque de 16×16 tendrá 16 píxeles en una dirección vertical ($y = 16$) y 16 píxeles en una dirección horizontal ($x = 16$). Asimismo, un bloque de tamaño $N \times N$ tiene, en general, N píxeles en una dirección vertical y N píxeles en una dirección horizontal, donde N representa un valor entero no negativo. Los píxeles de un bloque se pueden disponer en filas y columnas. Además, no es necesario que los bloques tengan necesariamente el mismo número de píxeles en la dirección horizontal y en la dirección vertical. Por ejemplo, los bloques pueden comprender $N \times M$ píxeles, donde M no es necesariamente igual a N.

[0058] Tras la codificación intrapredictiva o interpredictiva usando las PU de una CU, el codificador de vídeo 20 puede calcular datos residuales para las TU de la CU. Las PU pueden comprender datos sintácticos que describen un procedimiento o modo de generación de datos de píxeles predictivos en el dominio espacial (también denominado el dominio de píxeles) y las TU pueden comprender coeficientes en el dominio de transformación, tras la aplicación de una transformación, por ejemplo, una transformación de coseno discreta (DCT), una transformación entera, una transformación de ondículas o una transformación conceptualmente similar, en los datos de vídeo residuales. Los datos residuales pueden corresponder a diferencias de píxeles entre píxeles de la imagen no codificada y valores de predicción correspondientes a las PU. El codificador de vídeo 20 puede formar las TU, incluidos los datos residuales para la CU y, a continuación, transformar las TU para generar coeficientes de transformación para la CU.

[0059] Tras cualquier transformación para generar coeficientes de transformación, el codificador de vídeo 20 puede realizar la cuantificación de los coeficientes de transformación. La cuantificación es un término amplio que pretende tener su significado ordinario más amplio. En un modo de realización, la cuantificación se refiere a un proceso en el que unos coeficientes de transformación se cuantifican para reducir posiblemente la cantidad de datos usados para representar los coeficientes, proporcionando compresión adicional. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. Por ejemplo, un valor de n bits puede redondearse a la baja hasta un valor de m bits durante la cuantificación, donde n es mayor que m.

[0060] Después de la cuantificación, el codificador de vídeo puede explorar los coeficientes de transformación, produciendo un vector unidimensional a partir de la matriz bidimensional que incluye los coeficientes de transformación cuantificados. La exploración puede estar diseñada para colocar los coeficientes de energía más alta (y por lo tanto de menor frecuencia) al frente de la formación y para colocar los coeficientes de energía más baja (y por lo tanto de mayor frecuencia) al final de la formación. En algunos ejemplos, el codificador de vídeo 20 puede utilizar un orden de exploración predefinido para explorar los coeficientes de transformación cuantificados para producir un vector serializado que se pueda codificar por entropía. En otros ejemplos, el codificador de vídeo 20 puede realizar una exploración adaptativa. Después de explorar los coeficientes de transformación cuantificados para formar un vector unidimensional, el codificador de vídeo 20 puede realizar la codificación por entropía del vector unidimensional, por ejemplo, de acuerdo con la codificación de longitud variable adaptable al contexto (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación aritmética binaria adaptable al contexto basada en la sintaxis (SBAC), la codificación por entropía por división en intervalos de probabilidad (PIPE) o con otra metodología de codificación por entropía. El codificador de vídeo 20 también puede realizar la codificación por entropía de los elementos sintácticos asociados a los datos de vídeo codificados para su uso por el descodificador de vídeo 30 en la descodificación de los datos de vídeo.

[0061] Para realizar la CABAC, el codificador de vídeo 20 puede asignar un contexto, dentro de un modelo de contexto, a un símbolo a transmitir. El contexto se puede referir, por ejemplo, a si los valores vecinos del símbolo son distintos de cero o no. Para realizar la CAVLC, el codificador de vídeo 20 puede seleccionar un código de longitud variable para un símbolo a transmitir. Las palabras de código en la VLC se pueden generar de modo que los códigos relativamente más cortos correspondan a símbolos más probables, mientras que los códigos más largos correspondan a símbolos menos probables. De esta forma, el uso de la VLC puede lograr un ahorro en bits con respecto, por ejemplo, al uso de palabras de código de igual longitud para cada símbolo a transmitir. La determinación de la probabilidad se puede basar en un contexto asignado al símbolo.

[0062] Además, el codificador de vídeo 20 puede enviar datos sintácticos, tales como datos sintácticos basados en bloques, datos sintácticos basados en tramas y datos sintácticos basados en GOP, al descodificador de vídeo 30, por ejemplo, en una cabecera de trama, una cabecera de bloque, una cabecera de fragmento o una cabecera de GOP. Los datos sintácticos de GOP pueden describir una pluralidad de tramas en el GOP respectivo, y los datos sintácticos de trama pueden indicar un modo de codificación/predicción utilizado para codificar la trama correspondiente.

[0063] La FIG. 2 ilustra un flujo de proceso de ejemplo que puede ser implementado por varios codificadores de vídeo, como el codificador de vídeo 20 de la FIG. 3, el descodificador de vídeo 30 de la FIG. 4, o algún otro codificador de vídeo. En la siguiente descripción, para una mayor simplicidad, la textura reconstruida en el bloque de capa base se denomina predictor de BL (denotado como P_b). El predictor formado en la capa de mejora utilizando información de los bloques de capa de mejora se denomina predictor de EL (denotado como P_e). La predicción formada por la combinación adaptativa del predictor de BL y del predictor de EL se denomina predictor final (denotado como P_f) que es el que realmente se usa para predecir un bloque de capa de mejora actual, de acuerdo con las técnicas de esta divulgación. El proceso ilustrado en la FIG. 2 se puede usar para combinar predictores BL y predictores EL cuando se generan predicciones entre capas.

[0064] La información de vídeo de capa base 300 y la información de vídeo de capa de mejora 302 pueden obtenerse mediante un codificador de vídeo. En el bloque 310, puede comenzar la codificación de vídeo de capa base, tal como un bloque de BL. Un predictor de BL (por ejemplo, el bloque reconstruido en la capa base) puede generarse de acuerdo con la predicción de movimiento compensado, la intrapredicción y las técnicas de predicción residual descritas en el presente documento. En el bloque 312, también puede comenzar la codificación de vídeo de capa de mejora, tal como un bloque de EL. Se puede generar un predictor de EL de acuerdo con la predicción de movimiento compensado y la intrapredicción. Además, un predictor de BL generado en el bloque 310 puede combinarse con un predictor de EL generado en el bloque 312 para generar un predictor de EL final. Al generar el predictor de EL final se pueden aplicar factores de ponderación al predictor de BL y al predictor de EL para dar más peso al predictor que es más probable que sea preciso, como se describe a continuación.

[0065] En algunos modos de realización, el primer factor de ponderación puede ser constante para todos los píxeles del predictor de capa base, y el segundo factor de ponderación puede ser constante para todos los píxeles del predictor de capa base. En modos de realización adicionales, el primer factor de ponderación puede variar en función de la ubicación de píxel y el segundo factor de ponderación puede variar en función de la ubicación de píxel. En modos de realización adicionales, los factores de ponderación pueden variar en función de los parámetros de cuantificación o las relaciones de escalamiento espacial. En cualquiera de los ejemplos dados, los factores de ponderación pueden determinarse de tal manera que la suma del primer factor de ponderación y del segundo factor de ponderación sea igual a una constante.

[0066] Por ejemplo, en los píxeles de un borde o cerca del borde del bloque de capa de mejora actual (por ejemplo, el bloque de EL que se está codificando actualmente), el segundo factor de ponderación, correspondiente al predictor de EL, puede ser mayor que el primer factor de ponderación, correspondiente al predictor de BL. Dicha ponderación se puede usar en casos en los que los píxeles adyacentes o los píxeles próximos a un píxel del bloque de EL actual proporcionan mejores predictores que los píxeles de una capa base. En los píxeles que no están en un borde ni cerca del borde del bloque de capa de mejora actual, el primer factor de ponderación, correspondiente al predictor de BL, puede ser mayor que el segundo factor de ponderación, que corresponde al predictor de EL. En estos casos, se puede determinar que los píxeles del bloque de capa base que corresponden al bloque de capa de mejora actual proporcionan mejores predictores que los píxeles de otros bloques de capa de mejora. La determinación de cuándo un píxel está lo suficientemente cerca de un bloque de capa de mejora adyacente como para afectar a la ponderación aplicada a los predictores puede determinarse por el codificador de vídeo en función de una dirección de intrapredicción para el bloque de capa de mejora actual.

[0067] El codificador de vídeo también puede determinar un primer factor de ponderación en función de la distancia entre un píxel y un borde del bloque de capa de mejora actual. El segundo factor de ponderación puede aumentar, en general, para píxeles más cercanos al borde, y el primer factor de ponderación aumenta, en general, para píxeles más alejados del borde.

[0068] El codificador de vídeo también puede determinar el primer factor de ponderación y el segundo factor de ponderación basándose en un parámetro de cuantificación del predictor de capa base y un parámetro de cuantificación del predictor de capa de mejora. Por ejemplo, si el parámetro de cuantificación de la BL es sustancialmente igual a, o está dentro de algún umbral de, el parámetro de cuantificación de la EL, entonces los pesos del predictor de BL y del predictor de EL pueden ser más parecidos que si la diferencia en los parámetros de cuantificación de EL y BS es grande. Si el parámetro de cuantificación de BL es mucho mayor que el parámetro de cuantificación de EL, entonces el peso que el codificador de vídeo asigna al predictor de EL puede ser sustancialmente mayor que el peso asignado al predictor de BL (por ejemplo, el peso asignado al predictor de EL aumenta a medida que la diferencia $QP1 - QP2$ aumenta, donde $QP1$ representa el parámetro de cuantificación de la BL y $QP2$ representa el parámetro de cuantificación de la EL). Asimismo, si el parámetro de cuantificación de

BL es mucho más pequeño que el parámetro de cuantificación de EL, entonces el peso que el codificador de vídeo asigna al predictor de EL puede ser sustancialmente menor que el peso asignado al predictor de BL.

5 **[0069]** El codificador de vídeo también puede seleccionar el primer factor de ponderación y el segundo factor de ponderación en función de una relación de escalamiento. Por ejemplo, mayores relaciones de escalamiento pueden corresponder, en general, a valores más altos para el segundo factor de ponderación.

10 **[0070]** En el bloque 320 se pueden codificar datos residuales que representan las diferencias de píxeles entre el bloque de BL original y el bloque de BL predictivo generado como se describió anteriormente. Por ejemplo, durante la codificación de vídeo, los datos residuales pueden transformarse desde el dominio de píxel a un dominio de transformación, dando como resultado coeficientes de transformación residuales, que a continuación se pueden cuantificar. En el bloque 322 se pueden codificar datos residuales que representan diferencias de píxeles entre el bloque de EL original y el bloque de EL predictivo generados como se describe anteriormente.

15 **[0071]** En el bloque 330 se puede generar un predictor de un vector de movimiento para la capa base de acuerdo con técnicas de predicción de movimiento, tales como las descritas en el presente documento. En el bloque 332 también se puede generar un predictor de un vector de movimiento para la capa de mejora de acuerdo con las técnicas de predicción de movimiento. Además, el vector de movimiento de capa base se puede usar como un predictor para el vector de movimiento de capa de mejora.

20 **[0072]** En el bloque 340, el codificador de vídeo puede emitir vídeo de BL codificado. En el bloque 342, el codificador de vídeo puede emitir vídeo de EL codificado. Cuando el proceso se realiza mediante un codificador de vídeo, los residuos de BL y otros elementos sintácticos pueden codificarse por entropía para reducir aún más el tamaño del flujo de bits resultante, y los residuos de EL y otros elementos sintácticos pueden codificarse por entropía de manera similar. Después, los flujos de bits correspondientes a cada capa se pueden multiplexar para su salida (por ejemplo, transmisión o almacenamiento). Cuando el proceso se realiza mediante un descodificador de vídeo, el vídeo de EL codificado y el vídeo de BL pueden usarse para emitir una señal de vídeo para su presentación (por ejemplo, a través de una pantalla asociada al descodificador).

25 **[0073]** La FIG. 3 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo 20 que puede implementar técnicas de codificación de acuerdo con los aspectos descritos en esta divulgación. El codificador de vídeo 20 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. Como ejemplo, la unidad de selección de modo 40 puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación, tal como determinar los pesos de BL y EL en función de las ubicaciones de píxel, las características de BL y EL, y similares. Sin embargo, los aspectos de esta divulgación no se limitan a lo anterior. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del codificador de vídeo 20. En algunos ejemplos, de manera adicional o alternativa, un procesador (no mostrado) puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación.

30 **[0074]** El codificador de vídeo 20 puede realizar la intracodificación y la intercodificación de bloques de vídeo dentro de fragmentos de vídeo. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo dentro de una trama de vídeo o imagen dada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas o imágenes adyacentes de una secuencia de vídeo. El intramodo (modo I) puede referirse a cualquiera de varios modos de codificación de base espacial. Los intermodos, tales como la predicción unidireccional (modo P) o la bipredicción (modo B), pueden referirse a cualquiera de varios modos de codificación de base temporal.

35 **[0075]** Como se muestra en la FIG. 3, el codificador de vídeo 20 recibe un bloque de vídeo actual dentro de una trama de vídeo que se va a codificar. En el ejemplo de la FIG. 3, el codificador de vídeo 20 incluye una unidad de selección de modo 40, una memoria de tramas de referencia 64, un sumador 50, una unidad de procesamiento de transformación 52, una unidad de cuantificación 54 y una unidad de codificación por entropía 56. A su vez, la unidad de selección de modo 40 incluye una unidad de compensación de movimiento 44, una unidad de estimación de movimiento 42, una unidad de intrapredicción 46 y una unidad de división 48. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 20 incluye también una unidad de cuantificación inversa 58, una unidad de transformación inversa 60 y el sumador 62. También se puede incluir un filtro de eliminación de bloques (no mostrado en la FIG. 3) para filtrar fronteras de bloques, para eliminar distorsiones de efecto pixelado del vídeo reconstruido. Si se desea, el filtro de eliminación de bloques filtrará típicamente la salida del sumador 62. También pueden usarse filtros adicionales (en el bucle o tras el bucle), además del filtro de eliminación de bloques. Dichos filtros no se muestran por razones de brevedad pero, si se desea, pueden filtrar la salida del sumador 50 (como un filtro en el bucle).

40 **[0076]** Durante el proceso de codificación, el codificador de vídeo 20 recibe una trama o un fragmento de vídeo a codificar. La trama o el fragmento pueden dividirse en múltiples bloques de vídeo. La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 llevan a cabo la codificación interpredictiva del bloque de vídeo recibido con respecto a uno o más bloques en una o más tramas de referencia para proporcionar predicción temporal. La unidad de intrapredicción 46, de forma alternativa, puede llevar a cabo la codificación

intrapredicativa del bloque de vídeo recibido con respecto a uno o más bloques vecinos en la misma trama o fragmento que el bloque a codificar para proporcionar predicción espacial. El codificador de vídeo 20 puede llevar a cabo múltiples pasadas de codificación, por ejemplo, para seleccionar un modo de codificación adecuado para cada bloque de datos de vídeo.

5

[0077] Además, la unidad de división 48 puede dividir bloques de datos de vídeo en subbloques, basándose en la evaluación de los anteriores esquemas de división en las anteriores pasadas de codificación. Por ejemplo, la unidad de división 48 puede dividir inicialmente una trama o un fragmento en varias LCU, y dividir cada una de las LCU en varias sub-CU, basándose en un análisis de velocidad-distorsión (por ejemplo, una optimización de velocidad-distorsión). La unidad de selección de modo 40 puede producir además una estructura de datos de árbol cuaternario, indicativa de la división de una LCU en las sub-CU. Las CU de nodos hoja del árbol cuaternario pueden incluir una o más PU y una o más TU.

10

[0078] La unidad de selección de modo 40 puede seleccionar uno de los modos de codificación, intra o inter, por ejemplo, basándose en los resultados de errores, y proporciona el bloque intracodificado o intercodificado resultante al sumador 50 para generar datos de bloques residuales, y al sumador 62 para reconstruir el bloque codificado para su uso como una trama de referencia. La unidad de selección de modo 40 también proporciona elementos sintácticos, tales como vectores de movimiento, indicadores de intramodo, información de división y otra información sintáctica de este tipo, a la unidad de codificación por entropía 56.

15

20

[0079] La unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden estar sumamente integradas, pero se ilustran por separado con fines conceptuales. La estimación del movimiento, realizada por la unidad de estimación de movimiento 42, es el proceso de generación de vectores de movimiento, que estiman el movimiento para los bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de una PU de un bloque de vídeo dentro de una trama o imagen de vídeo actual, con respecto a un bloque predictivo dentro de una trama de referencia (u otra unidad codificada), con respecto al bloque actual que se está codificando dentro de la trama actual (u otra unidad codificada). Un bloque predictivo es un bloque que se revela como estrechamente coincidente con el bloque a codificar, en lo que respecta a la diferencia de píxeles, lo cual puede determinarse mediante la suma de diferencias absolutas (SAD), suma de las diferencias al cuadrado (SSD) u otras métricas de diferencia. En algunos ejemplos, el codificador de vídeo 20 puede calcular valores para posiciones fraccionarias de píxeles de imágenes de referencia almacenadas en la memoria de tramas de referencia 64. Por ejemplo, el codificador de vídeo 20 puede interpolar valores de posiciones de un cuarto de píxel, posiciones de un octavo de píxel u otras posiciones fraccionarias de píxel de la imagen de referencia. Por lo tanto, la unidad de estimación de movimiento 42 puede realizar una búsqueda de movimiento relativa a las posiciones de píxel completo y las posiciones de píxel fraccionario, y proporcionar un vector de movimiento con una precisión de píxel fraccionario.

25

30

35

[0080] La unidad de estimación de movimiento 42 calcula un vector de movimiento para una PU de un bloque de vídeo en un fragmento intercodificado, comparando la posición de la PU con la posición de un bloque predictivo de una imagen de referencia. La imagen de referencia puede seleccionarse a partir de una primera lista de imágenes de referencia (Lista 0) o una segunda lista de imágenes de referencia (Lista 1), cada una de las cuales identifica una o más imágenes de referencia almacenadas en la memoria de tramas de referencia 64. La unidad de estimación de movimiento 42 envía el vector de movimiento calculado a la unidad de codificación por entropía 56 y a la unidad de compensación de movimiento 44.

40

45

[0081] La compensación de movimiento, llevada a cabo por la unidad de compensación de movimiento 44, puede implicar capturar o generar el bloque predictivo basándose en el vector de movimiento determinado por la unidad de estimación de movimiento 42. De nuevo, la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44 pueden integrarse funcionalmente, en algunos ejemplos. Tras recibir el vector de movimiento para la PU del bloque de vídeo actual, la unidad de compensación de movimiento 44 puede localizar el bloque predictivo al que apunta el vector de movimiento en una de las listas de imágenes de referencia. El sumador 50 forma un bloque de vídeo residual restando los valores de píxel del bloque predictivo a los valores de píxel del bloque de vídeo actual que se está codificando, generando valores de diferencias de píxel, como se analiza posteriormente. En general, la unidad de estimación de movimiento 42 lleva a cabo la estimación de movimiento con respecto a los componentes de luminancia, y la unidad de compensación de movimiento 44 utiliza los vectores de movimiento calculados basándose en los componentes de luminancia, tanto para los componentes de crominancia como para los componentes de luminancia. La unidad de selección de modo 40 también puede generar elementos sintácticos asociados a los bloques de vídeo y el fragmento de vídeo para su uso mediante el decodificador de vídeo 30 en la decodificación de los bloques de vídeo del fragmento de vídeo.

50

55

60

[0082] La unidad de intrapredicción 46 puede intrapredicir o calcular un bloque actual, como alternativa a la interpredicción llevada a cabo por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como se ha descrito anteriormente. En particular, la unidad de intrapredicción 46 puede determinar un modo de intrapredicción a usar para codificar un bloque actual. En algunos ejemplos, la unidad de intrapredicción 46 puede codificar un bloque actual usando diversos modos de intrapredicción, por ejemplo, durante pasadas de codificación independientes, y la unidad de intrapredicción 46 (o la unidad de selección de modo 40,

65

en algunos ejemplos) puede seleccionar un modo de intrapredicción adecuado a usar a partir de los modos probados.

5 **[0083]** Por ejemplo, la unidad de intrapredicción 46 puede calcular valores de velocidad-distorsión usando un análisis de velocidad-distorsión para los diversos modos de intrapredicción probados, y seleccionar el modo de intrapredicción que tenga las mejores características de velocidad-distorsión entre los modos probados. El análisis de velocidad-distorsión determina, en general, una cantidad de distorsión (o de errores) entre un bloque codificado y un bloque original, no codificado, que se codificó para producir el bloque codificado, así como una velocidad de transmisión de bits (es decir, un número de bits) usada para producir el bloque codificado. La unidad de intrapredicción 46 puede calcular valores de coste a partir de las distorsiones y velocidades para los diversos bloques codificados para determinar qué modo de intrapredicción presenta el mejor valor de velocidad-distorsión (o el valor de coste más bajo) para el bloque.

15 **[0084]** Después de seleccionar un modo de intrapredicción para un bloque, la unidad de intrapredicción 46 puede proporcionar información, indicativa del modo de intrapredicción seleccionado para el bloque, a la unidad de codificación por entropía 56. La unidad de codificación por entropía 56 puede codificar la información que indica el modo de intrapredicción seleccionado. El codificador de vídeo 20 puede incluir, en el flujo de bits transmitido, datos de configuración que pueden incluir una pluralidad de tablas de índices de modo de intrapredicción y una pluralidad de tablas modificadas de índices de modo de intrapredicción (también denominadas tablas de correlación de palabras de código), definiciones de contextos de codificación para diversos bloques e indicaciones de un modo de intrapredicción más probable, una tabla de índices de modo de intrapredicción y una tabla modificada de índices de modo de intrapredicción a usar para cada uno de los contextos.

25 **[0085]** El codificador de vídeo 20 forma un bloque de vídeo residual restando los datos de predicción de la unidad de selección de modo 40 del bloque de vídeo original que se está codificando. El sumador 50 representa el componente o los componentes que realizan esta operación de resta. La unidad de procesamiento de transformación 52 aplica una transformación, tal como una transformación discreta de coseno (DCT) o una transformación conceptualmente similar, al bloque residual, produciendo un bloque de vídeo que comprende valores residuales de coeficientes de transformación. La unidad de procesamiento de transformación 52 puede llevar a cabo otras transformaciones que son conceptualmente similares a la DCT. También se podrían usar transformaciones de ondículas, transformaciones de números enteros, transformaciones de subbandas u otros tipos de transformaciones. En cualquier caso, la unidad de procesamiento de transformación 52 aplica la transformación al bloque residual, produciendo un bloque de coeficientes de transformación residuales. La transformación puede convertir la información residual desde un dominio de valores de píxel a un dominio de transformación, tal como un dominio de frecuencia. La unidad de procesamiento de transformación 52 puede enviar los coeficientes de transformación resultantes a la unidad de cuantificación 54. La unidad de cuantificación 54 cuantifica los coeficientes de transformación para reducir más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos, o a la totalidad, de los coeficientes. El grado de cuantificación se puede modificar ajustando un parámetro de cuantificación. En algunos ejemplos, la unidad de cuantificación 54 puede realizar, a continuación, una exploración de la matriz que incluye los coeficientes de transformación cuantificados. De forma alternativa, la unidad de codificación por entropía 56 puede realizar la exploración.

45 **[0086]** Tras la cuantificación, la unidad de codificación por entropía 56 codifica por entropía los coeficientes de transformación cuantificados. Por ejemplo, la unidad de codificación por entropía 56 puede llevar a cabo la codificación de longitud variable adaptable al contexto (CAVLC), la codificación aritmética binaria adaptable al contexto (CABAC), la codificación aritmética binaria adaptable al contexto basada en sintaxis (SBAC), la codificación por entropía por división en intervalos de probabilidad (PIPE) u otra técnica de codificación por entropía. En el caso de la codificación por entropía basada en el contexto, el contexto se puede basar en bloques vecinos. Tras la codificación por entropía por parte de la unidad de codificación por entropía 56, el flujo de bits codificado se puede transmitir a otro dispositivo (por ejemplo, el descodificador de vídeo 30) o archivar para su posterior transmisión o recuperación.

55 **[0087]** La unidad de cuantificación inversa 58 y la unidad de transformación inversa 60 aplican la cuantificación inversa y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxel, por ejemplo, para su uso posterior como bloque de referencia. La unidad de compensación de movimiento 44 puede calcular un bloque de referencia añadiendo el bloque residual a un bloque predictivo de una de las tramas de la memoria de tramas de referencia 64. La unidad de compensación de movimiento 44 también puede aplicar uno o más filtros de interpolación al bloque residual reconstruido para calcular valores fraccionarios de píxel para su uso en la estimación de movimiento. El sumador 62 añade el bloque residual reconstruido al bloque de predicción compensado por movimiento, generado por la unidad de compensación de movimiento 44, para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria de tramas de referencia 64. El bloque de vídeo reconstruido puede ser utilizado por la unidad de estimación de movimiento 42 y la unidad de compensación de movimiento 44, como bloque de referencia para intercodificar un bloque en una trama de vídeo posterior.

65

[0088] De esta manera, el codificador de vídeo 20 de la FIG. 3 representa un ejemplo de un codificador de vídeo configurado para realizar las técnicas de esta divulgación. Por ejemplo, el codificador de vídeo 20 puede combinar un predictor de capa base y un predictor de capa de mejora para formar un predictor final, y realizar una predicción entre capas para un bloque de capa de mejora actual basado en el predictor final. Realizar una predicción entre capas puede incluir realizar una predicción de textura entre capas. Al combinar el predictor de capa base y el predictor de capa de mejora, el codificador de vídeo 20 puede asignar un primer factor de ponderación al predictor de capa base y asignar un segundo factor de ponderación al predictor de capa de mejora.

[0089] La FIG. 4 es un diagrama de bloques que ilustra un ejemplo de un descodificador de vídeo que puede implementar técnicas de acuerdo con aspectos descritos en esta divulgación. El descodificador de vídeo 30 puede configurarse para realizar algunas de, o todas, las técnicas de esta divulgación. Como ejemplo, la unidad de compensación de movimiento 72 y/o la unidad de intrapredicción 74 pueden configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación, como la determinación de los pesos de BL y EL según las ubicaciones de píxel, las características de BL y EL, y similares. Sin embargo, los aspectos de esta divulgación no se limitan a lo anterior. En algunos ejemplos, las técnicas descritas en esta divulgación pueden compartirse entre los diversos componentes del descodificador de vídeo 30. En algunos ejemplos, de manera adicional o alternativa, un procesador (no mostrado) puede configurarse para realizar cualquiera o todas las técnicas descritas en esta divulgación.

[0090] En el ejemplo de la FIG. 4, el descodificador de vídeo 30 incluye una unidad de descodificación por entropía 70, una unidad de compensación de movimiento 72, una unidad de intrapredicción 74, una unidad de cuantificación inversa 76, una unidad de transformación inversa 78, una memoria de tramas de referencia 82 y un sumador 80. En algunos ejemplos, el descodificador de vídeo 30 puede llevar a cabo una pasada de descodificación en general recíproca a la pasada de codificación descrita con respecto al codificador de vídeo 20 (FIG. 3A). La unidad de compensación de movimiento 72 puede generar datos de predicción basándose en vectores de movimiento recibidos desde la unidad de descodificación por entropía 70, mientras que la unidad de intrapredicción 74 puede generar datos de predicción basándose en indicadores de modo de intrapredicción, recibidos desde la unidad de descodificación por entropía 70.

[0091] Durante el proceso de descodificación, el descodificador de vídeo 30 recibe un flujo de bits de vídeo codificado, que representa bloques de vídeo de un fragmento de vídeo codificado y elementos sintácticos asociados, desde el codificador de vídeo 20. La unidad de descodificación por entropía 70 del descodificador de vídeo 30 descodifica por entropía el flujo de bits para generar coeficientes cuantificados, vectores de movimiento o indicadores de modo de intrapredicción y otros elementos sintácticos. La unidad de descodificación por entropía 70 reenvía los vectores de movimiento y otros elementos sintácticos a la unidad de compensación de movimiento 72. El descodificador de vídeo 30 puede recibir los elementos sintácticos en el nivel de fragmento de vídeo y/o el nivel de bloque de vídeo.

[0092] Cuando el fragmento de vídeo se codifica como un fragmento intracodificado (I), la unidad de intrapredicción 74 puede generar datos de predicción para un bloque de vídeo del fragmento de vídeo actual basándose en un modo de intrapredicción señalado y en datos de bloques previamente descodificados de la trama o imagen actual. Cuando la trama de vídeo se codifica como un fragmento intercodificado (por ejemplo, B, P o GPB), la unidad de compensación de movimiento 72 genera bloques predictivos para un bloque de vídeo del fragmento de vídeo actual basándose en los vectores de movimiento y en otros elementos sintácticos recibidos desde la unidad de descodificación por entropía 70. Los bloques predictivos se pueden generar a partir de una de las imágenes de referencia dentro de una de las listas de imágenes de referencia. El descodificador de vídeo 30 puede construir las listas de tramas de referencia, la Lista 0 y la Lista 1, usando técnicas de construcción por omisión, basándose en las imágenes de referencia almacenadas en la memoria de imágenes de referencia 92. La unidad de compensación de movimiento 72 determina la información de predicción para un bloque de vídeo del fragmento de vídeo actual, analizando sintácticamente los vectores de movimiento y otros elementos sintácticos, y usa la información de predicción para generar los bloques predictivos para el bloque de vídeo actual que se está descodificando. Por ejemplo, la unidad de compensación de movimiento 72 usa algunos de los elementos sintácticos recibidos para determinar un modo de predicción (por ejemplo, intrapredicción o interpredicción) usada para codificar los bloques de vídeo del fragmento de vídeo, un tipo de fragmento de interpredicción (por ejemplo, fragmento B, fragmento P o fragmento GPB), información de construcción para una o más de las listas de imágenes de referencia para el fragmento, vectores de movimiento para cada bloque de vídeo intercodificado del fragmento, el estado de interpredicción para cada bloque de vídeo intercodificado del fragmento y otra información para descodificar los bloques de vídeo en el fragmento de vídeo actual.

[0093] La unidad de compensación de movimiento 72 también puede realizar la interpolación basándose en filtros de interpolación. La unidad de compensación de movimiento 72 puede usar filtros de interpolación como los usados por el codificador de vídeo 20 durante la codificación de los bloques de vídeo para calcular valores interpolados para píxeles fraccionarios de bloques de referencia. En este caso, la unidad de compensación de movimiento 72 puede determinar los filtros de interpolación usados por el codificador de vídeo 20 a partir de los elementos sintácticos recibidos y usar los filtros de interpolación para generar bloques predictivos.

5 [0094] La unidad de cuantificación inversa 76 cuantifica de manera inversa, por ejemplo, descuantifica, los coeficientes de transformación cuantificados, proporcionados en el flujo de bits y descodificados por la unidad de descodificación por entropía 80. El proceso de cuantificación inversa puede incluir el uso de un parámetro de cuantificación QPy, calculado por el descodificador de vídeo 30 para cada bloque de vídeo en el fragmento de vídeo, para determinar un grado de cuantificación y, asimismo, un grado de cuantificación inversa que debería aplicarse.

10 [0095] La unidad de transformación inversa 78 aplica una transformación inversa, por ejemplo, una DCT inversa, una transformación entera inversa, o un proceso de transformación inversa conceptualmente similar, a los coeficientes de transformación, con el fin de generar bloques residuales en el dominio de píxel.

15 [0096] Después de que la unidad de compensación de movimiento 82 genere el bloque predictivo para el bloque de vídeo actual basándose en los vectores de movimiento y otros elementos sintácticos, el descodificador de vídeo 30 forma un bloque de vídeo descodificado sumando los bloques residuales de la unidad de transformación inversa 78 a los correspondientes bloques predictivos generados por la unidad de compensación de movimiento 72. El sumador 90 representa el componente o los componentes que realizan esta operación de suma. Si se desea, también se puede aplicar un filtro de eliminación de bloques para filtrar los bloques descodificados a fin de eliminar distorsiones de efecto pixelado. También se pueden usar otros filtros de bucle (en el bucle de codificación o bien después del bucle de codificación) para allanar las transiciones de píxeles o mejorar de otro modo la calidad del vídeo. Los bloques de vídeo descodificados de una trama o imagen dada se almacenan a continuación en la memoria de imágenes de referencia 92, que almacena imágenes de referencia usadas para una compensación de movimiento subsiguiente. La memoria de tramas de referencia 82 almacena también vídeo descodificado para su presentación posterior en un dispositivo de visualización, tal como el dispositivo de visualización 32 de la FIG. 1.

25 [0097] De esta manera, el descodificador de vídeo 30 de la FIG. 4 representa un ejemplo de un descodificador de vídeo configurado para realizar las técnicas de esta divulgación. Por ejemplo, el descodificador de vídeo 30 puede combinar un predictor de capa base y un predictor de capa de mejora para formar un predictor final, y realizar una predicción entre capas para un bloque de capa de mejora actual basado en el predictor final. Realizar una predicción entre capas puede incluir realizar una predicción de textura entre capas. Al combinar el predictor de capa base y el predictor de capa de mejora, el descodificador de vídeo 30 puede asignar un primer factor de ponderación al predictor de capa base y asignar un segundo factor de ponderación al predictor de capa de mejora. En algunos casos, el primer factor de ponderación puede ser constante para todos los píxeles del predictor de capa base, y el segundo factor de ponderación puede ser constante para todos los píxeles del predictor de capa base.

35 [0098] El descodificador de vídeo 30 puede determinar los factores de ponderación de, entre otras cosas, las características de los píxeles dentro de los bloques de capa de mejora que se descodifican y las características de los predictores de BL y los predictores de EL. El descodificador de vídeo 30 puede estar configurado para determinar los factores de ponderación de una manera similar a la del codificador de vídeo 20, descrito anteriormente. De esta manera, el descodificador de vídeo 30 puede determinar los mismos o casi los mismos parámetros de ponderación que se usaron para codificar bloques de vídeo sin requerir datos adicionales, como los datos referentes a qué pesos usar.

45 [0099] Como se describió anteriormente con respecto al codificador de vídeo 20, el primer factor de ponderación puede variar en función de la ubicación de píxel y el segundo factor de ponderación puede variar en función de la ubicación de píxel. Por ejemplo, para píxeles en un borde o cerca del borde del bloque de capa de mejora actual, el segundo factor de ponderación podría ser mayor que el primer factor de ponderación. Para los píxeles que no están en el borde ni cerca del borde del bloque de capa de mejora actual, el primer factor de ponderación puede ser mayor que el segundo factor de ponderación. La determinación de lo que se constituye cerca del borde puede determinarse mediante el descodificador de vídeo 30 en función de una dirección de intrapredicción para el bloque de capa de mejora actual.

50 [0100] El descodificador de vídeo 30 también puede determinar un primer factor de ponderación en función de la distancia entre un píxel y un borde del bloque de mejora actual. El segundo factor de ponderación puede aumentar, en general, para píxeles más cercanos al borde, y el primer factor de ponderación aumenta, en general, para píxeles más alejados del borde.

60 [0101] El descodificador de vídeo 30 también puede determinar el primer factor de ponderación y el segundo factor de ponderación en base a un parámetro de cuantificación del predictor de capa base y un parámetro de cuantificación del predictor de capa de mejora de una manera similar a la descrita anteriormente con respecto a la FIG. 3.

65 [0102] El descodificador de vídeo 30 también puede seleccionar el primer factor de ponderación y el segundo factor de ponderación basándose una relación de escalamiento. Por ejemplo, mayores relaciones de escalamiento pueden corresponder, en general, a valores más altos para el segundo factor de ponderación.

[0103] Volviendo ahora a la FIG. 5, a continuación se describe un ejemplo de predicción espacial (es decir, intrapredicción) utilizada en un codificador H.264. Para realizar la predicción espacial, el codificador H.264 ofrece 9 modos para la predicción de bloques 4x4, incluida la predicción DC (Modo 0) y 8 modos direccionales, etiquetados del 1 al 8, como se muestra en la FIG. 5. El proceso de predicción se ilustra en la FIG. 6. Como se muestra en la FIG. 6, los píxeles de *a* a *p* en el bloque 600 se van a codificar, y los píxeles de **A** a **R** de bloques vecinos 602-608 ya se han codificado y se pueden utilizar para la predicción. Si, por ejemplo, se selecciona el modo 1, entonces los píxeles *a*, *e*, *i* y *m* se predicen estableciéndolos igual al píxel **A**, y los píxeles *b*, *f*, *j* y *n* se predicen estableciéndolos igual al píxel **B**, etc. De manera similar, si se selecciona el modo 2, los píxeles *a*, *b*, *c* y *d* se predicen estableciéndolos igual al píxel **I**, y los píxeles *e*, *f*, *g* y *h* se predicen estableciéndolos igual al píxel **J**, etc. Por lo tanto, en este ejemplo, el modo 1 es un predictor en la dirección vertical; y el modo 2 es un predictor en la dirección horizontal.

[0104] Las definiciones de intramodo y los ejemplos ilustrados en las FIG. 5 y 6 son solo ilustrativos y no pretenden ser limitativos. Se pueden usar otras definiciones de intramodo, como las de HEVC. En HEVC, el número de modos de predicción aumenta aún más. Por ejemplo, hay 35 modos de predicción disponibles, respectivamente, para bloques de 4x4, bloques de 8x8 y más grandes. El índice de modo definido para cada modo en HEVC también puede ser diferente. Por ejemplo, en HEVC, la predicción plana se define como modo 0 y la predicción de DC como modo 1.

[0105] En la predicción de texturas entre capas como se describe en el intramodo de BL, la textura reconstruida de un bloque de capa base se puede usar como un predictor para predecir la textura de un bloque de capa de mejora correspondiente. Sin embargo, en ciertos casos, no todos los píxeles de la textura reconstruida a partir del bloque de capa base son el mejor predictor o el preferente para los píxeles correspondientes en el bloque de capa de mejora. Los esquemas actuales de la técnica no ofrecen la flexibilidad de usar solamente una parte de los píxeles o subponderar una parte de los píxeles a partir de una capa base para formar un predictor para un bloque de capa de mejora. Esta divulgación describe técnicas para permitir dicha flexibilidad para lograr una mejor eficiencia de codificación.

[0106] En algunos modos de realización, se puede usar una combinación ponderada constante de predictores. De acuerdo con dicha técnica, ciertas ponderaciones pueden asignarse a un predictor de EL, *Pe* y un predictor de BL, *Pb*, respectivamente, cuando se combinan para formar un predictor final *Pf*. Dichas ponderaciones pueden ser constantes para todos los píxeles dentro de un predictor. Por ejemplo, un predictor final puede formarse en base a la siguiente ecuación.

$$P_f = c_1 * P_b + c_2 * P_e \quad (1)$$

[0107] En la ecuación (1), *c1* y *c2* representan coeficientes de ponderación. La suma de *c1* y *c2* puede ser igual a una constante, por ejemplo, 1. Tanto *c1* como *c2* pueden permanecer constantes para todos los píxeles dentro de *Pb* y *Pe*.

[0108] En algunos modos de realización, en lugar de utilizar una ponderación constante, los coeficientes de ponderación para cada predictor pueden variar según la ubicación de los píxeles. Por ejemplo, los coeficientes de ponderación pueden ser diferentes para diferentes ubicaciones de píxel dentro de un predictor. Más específicamente, el predictor final se puede formar en base a la siguiente ecuación.

$$P_f[i, j] = c_1[i, j] * P_b[i, j] + c_2[i, j] * P_e[i, j] \quad (2)$$

[0109] En la ecuación (2), *i* y *j* representan coordenadas bidimensionales dentro de un predictor, que indican filas y columnas respectivamente. Los parámetros de ponderación *c1* y *c2* pueden ser diferentes en diferentes ubicaciones [*i,j*]. Sin embargo, la suma de *c1*[*i,j*] y *c2*[*i,j*] puede ser igual a una constante, como 1. Según la ecuación (2), si *c1*[*i,j*] = 1 y *c2*[*i,j*] = 0, el píxel correspondiente en el predictor final se forma totalmente a partir del píxel en el predictor de BL. De forma alternativa, si *c1*[*i,j*] = 0 y *c2*[*i,j*] = 1, entonces el píxel correspondiente en el predictor final se forma totalmente a partir del píxel en el predictor de EL.

[0110] En algunos modos de realización, cuando el predictor de EL se forma utilizando la intrapredicción, a los píxeles del predictor de EL que se encuentran en el borde del bloque de capa de EL actual (por ejemplo, el bloque que se está codificando actualmente) y cerca de un bloque vecino de píxeles reconstruidos se les puede dar una mayor ponderación cuando se combinan con el predictor de BL para formar el predictor final. Para otras ubicaciones de píxeles, se puede otorgar una mayor ponderación a los píxeles del predictor de BL.

[0111] Los píxeles reconstruidos en la capa de mejora suelen ser más precisos (por ejemplo, tienen un valor más cercano a sus valores de píxel originales) que los píxeles correspondientes en la capa base. Como resultado, en el predictor de EL debido a la fuerte correlación entre los píxeles vecinos, los píxeles ubicados cerca de los píxeles reconstruidos de un bloque vecino se pueden usar para generar predicciones de mayor calidad (por ejemplo, a los predictores de EL se les puede otorgar un peso mayor que el de los predictores de BL correspondientes). Sin embargo, en otros píxeles en el predictor de EL que se encuentran lejos de los píxeles reconstruidos de bloques

vecinos que se usan para la predicción, la calidad de la predicción puede ser menor en algunos casos. En estos casos, los predictores de BL pueden usarse para generar predictores finales de mayor calidad (por ejemplo, el predictor de BL puede tener un peso mayor que el de los predictores de EL correspondientes).

5 **[0112]** Esta idea se ilustra en la FIG. 7. En la FIG. 7, el bloque de EL 700 es el bloque de EL actual que se debe predecir y codificar. El área 720 (marcada con el patrón de línea "/") es el área de píxeles reconstruida de los bloques de EL vecinos 702-708 que pueden usarse para la predicción de un bloque de EL actual 700. El área 710, que se muestra sombreada, es el área cercana a los bloques de EL vecinos 702-708, donde una ponderación relativamente mayor se puede asignar al predictor de EL. Por ejemplo, según la ecuación (2), $c1[i,j]$ puede ser menor que 0,5 y $c2[i,j]$ mayor que 0,5 para los píxeles en el área sombreada 710. En los píxeles en el borde de los bloques de EL vecinos 702-708 dentro del área sombreada 710, $c1[i,j]$ puede tomar un valor de 0 y $c2[i,j]$ un valor de 1 para que el valor de predicción en el predictor final esté totalmente basado en el predictor de EL para estos píxeles.

15 **[0113]** El caso mostrado en la FIG. 7 es solo un ejemplo y no pretende ser limitante. En la práctica, la forma del área sombreada 710 puede definirse de forma adaptativa, por ejemplo, en función de una dirección de intrapredicción seleccionada para el bloque actual 700. Las FIG. 8A y 8B ilustran otros dos ejemplos en los que el área sombreada se puede definir de manera diferente. La FIG. 8A ilustra el caso en que una dirección de predicción es vertical. En este caso, se puede otorgar una mayor ponderación al predictor de EL para los píxeles en el área sombreada 810a sustancialmente en o cerca del borde de un bloque de EL vecino en la dirección vertical. La FIG. 8B muestra un ejemplo en el que una dirección de predicción es horizontal. En este caso, se puede otorgar una mayor ponderación al predictor de EL para los píxeles en el área sombreada 810b sustancialmente en o cerca del borde de un bloque de EL vecino en la dirección horizontal.

25 **[0114]** Aunque se indica un área sombreada en los ejemplos descritos anteriormente, los parámetros de ponderación aún pueden variar dentro o fuera del área sombreada. El área sombreada solo se usa como ejemplo para ilustrar una región en la que se puede otorgar más ponderación al predictor de EL que al predictor de BL para formar un predictor final. En algunos modos de realización, el factor de ponderación para una determinada ubicación de píxel puede determinarse en función de la dirección de intrapredicción y esencialmente de la distancia de la ubicación de píxel desde el píxel de borde correspondiente que se usa para la predicción. Por ejemplo, en distancias más cortas para una ubicación de píxel determinada, se pueden asignar factores de ponderación más altos al píxel correspondiente en el predictor de EL.

35 **[0115]** En modos de realización adicionales, se puede usar una combinación ponderada de predictores en función de parámetros de cuantificación o relaciones de escalamiento espacial. Al determinar los coeficientes (por ejemplo, los parámetros de ponderación) para combinar predictores de una capa base y una capa de mejora, también se puede considerar un parámetro de cuantificación (QP). En general, cuanto más bajo sea el valor de QP, mayor será la calidad de un predictor. Como resultado, si los bloques de capa de mejora tienen QP significativamente más bajos que los utilizados en el bloque de capa base, se puede asignar más ponderación al predictor de EL para formar el predictor final para un bloque de capa de mejora actual. Por ejemplo, la diferencia de QP entre el bloque de capa base y el bloque de capa de mejora se puede usar como un factor para determinar parámetros de ponderación.

45 **[0116]** También se puede considerar una relación de escalamiento espacial al determinar los parámetros de ponderación. En la escalabilidad espacial, las tramas de mejora tienen resoluciones espaciales más altas que las tramas de capa base. En este caso, la textura reconstruida de capa base se debe sobremuestrear en consecuencia antes de que se pueda usar como predictor para un bloque de capa de mejora correspondiente. La calidad de la textura puede degradarse durante el proceso de sobremuestreo. Mayores relaciones de escalamiento dan como resultado, en general, una mayor degradación en la calidad. Por consiguiente, la relación de escalamiento se puede considerar como un factor en la determinación de los parámetros de ponderación. En general, cuanto mayor sea la relación de escalamiento, más peso debe asignarse al predictor de EL para generar un predictor final.

55 **[0117]** En lo que respecta a las técnicas descritas hasta ahora, los parámetros de ponderación pueden determinarse u obtenerse implícitamente dependiendo de ciertas condiciones o contextos. De esta manera, una señalización adicional, de codificador a descodificador, para los parámetros de ponderación puede no ser necesaria cuando tanto el codificador como el descodificador utilizan el mismo esquema para obtener parámetros de ponderación.

60 **[0118]** Como se describió anteriormente, las condiciones o contextos utilizados para obtener implícitamente los parámetros de ponderación pueden incluir la dirección de intrapredicción (o intramodo), la distancia de una ubicación de píxel desde el píxel de borde correspondiente utilizado para la predicción, un QP de capa base y un QP de capa de mejora, así como la relación de escalamiento espacial entre diferentes capas. También se pueden usar otros factores, tales como las diferencias de vectores de movimiento, el tamaño de bloque, etc.

65 **[0119]** Como alternativa a la obtención implícita de los parámetros de ponderación, un codificador también puede señalar cierta información a un descodificador para determinar los parámetros de ponderación. Por ejemplo, un

codificador puede señalar coordenadas en el bloque actual para indicar cómo deben calcularse los parámetros de ponderación.

[0120] En la FIG. 9 se proporciona un ejemplo. Cuando la dirección de la intrapredicción es vertical, una coordenada de fila 910 puede ser señalizada a un decodificador que indica la ubicación donde se asigna una ponderación idéntica (por ejemplo, 0,5) entre el predictor de EL y el predictor de BL. Supongamos que la fila superior tiene una ponderación de 1 del predictor de EL (es decir, una ponderación de 0 de BL) y la fila inferior tiene una ponderación de 1 del predictor de BL (es decir, una ponderación de 0 de EL), los parámetros de ponderación para todas las demás ubicaciones de píxel dentro del bloque de capa de mejora actual 900 se pueden interpolar de acuerdo con un esquema determinado, por ejemplo, interpolación lineal. La misma técnica se puede aplicar cuando la dirección de predicción es horizontal. Tal idea puede extenderse a casos bidimensionales. En algunos modos de realización, una coordenada tanto de fila como de columna pueden ser señalizadas a un decodificador para que la interpolación de los parámetros de ponderación se realice de una manera bidimensional. En algunos modos de realización, la coordenada de fila y/o columna no indica el punto en el que la ponderación dada al predictor de BL y al predictor de EL es uniforme, sino que puede indicar el punto en el que la ponderación del predictor de BL o del predictor de EL es para formar la base completa para el predictor final. En tales casos, los pesos asignados al resto de las posiciones pueden inferirse de una manera similar.

[0121] Volviendo ahora a la FIG. 10, se describirá un proceso ilustrativo para la codificación de vídeo. El proceso puede implementarse en uno o más de los dispositivos descritos en el presente documento, tal como el dispositivo de origen 12 o el dispositivo de destino 14.

[0122] El proceso comienza en el bloque 1002, donde se recibe una primera capa del vídeo digital y una segunda capa del vídeo digital. Cuando se implementa mediante un dispositivo de destino 14 o algún otro decodificador, las capas pueden transmitirse como señales digitales a través de medios cableados y/o inalámbricos (por ejemplo, USB, PSTN, Ethernet, celular, satélite, WiFi, Bluetooth, etc.). Las capas se pueden recibir por separado o como parte de la misma transmisión. En algunos modos de realización, las capas pueden recibirse directamente desde el dispositivo de origen 12 o desde el dispositivo de origen a través de uno o más intermediarios. Las capas o partes de las mismas también pueden recibirse desde múltiples dispositivos de origen. El proceso de recepción también puede incluir el almacenamiento de las capas recibidas en memoria.

[0123] En algunos modos de realización, el proceso puede ser implementado por un dispositivo de origen 12 o algún otro codificador. En algunos casos, el vídeo digital que incluye una primera capa y una segunda capa se puede obtener de la memoria (por ejemplo, vídeo almacenado en un disco, DVD, Blu-ray, etc.). La memoria puede incluirse en el dispositivo que implementa el proceso, o la memoria puede ser un dispositivo individual (por ejemplo, almacenamiento en red) que sea accesible para el dispositivo que implementa el procedimiento. En algunos modos de realización, el vídeo digital se puede obtener de un sensor (por ejemplo, una cámara).

[0124] En el bloque 1004, el dispositivo que implementa el proceso puede determinar los pesos para predictores asociados a la primera capa (por ejemplo, predictores de capa base) y para predictores asociados a la segunda capa (por ejemplo, predictores de capa de mejora). Por ejemplo, un dispositivo de destino 14 puede estar descodificando un bloque de capa de mejora a partir de una trama particular del vídeo basándose en predictores de EL y predictores de BL ponderados. Un bloque de EL vecino puede haber sido ya descodificado. En cuanto a un píxel cercano al bloque vecino, el descodificador puede asignar un mayor peso al predictor de EL para ese píxel que al predictor de BL correspondiente (por ejemplo, el predictor ubicado en la capa base). En cuanto a un píxel que está lejos del bloque de EL vecino descodificado previamente, el descodificador puede asignar un mayor peso al predictor de BL correspondiente al píxel que al predictor de EL. A continuación, haciendo referencia a la FIG. 11, se describe un proceso ilustrativo para asignar pesos a los predictores de EL y BL en función de la ubicación de píxel. También se pueden considerar otros factores al determinar los pesos. Por ejemplo, puede considerarse la dirección de predicción. En tales casos, a los predictores de EL asociados a píxeles que están cerca de un bloque vecino, pero que no están alineados con la dirección de predicción actual, no se les tiene que asignar necesariamente un peso mayor que el predictor de BL correspondiente. Otros factores que pueden considerarse incluyen valores de QP, relaciones de escalamiento espacial y similares. A continuación, haciendo referencia a la FIG. 12, se describe un proceso ilustrativo para asignar pesos a los predictores de EL y BL en función de las características de EL y BL.

[0125] En el bloque 1006, el dispositivo que implementa el proceso puede codificar unidades de vídeo de capa de mejora (por ejemplo, bloques o píxeles) en función de los predictores ponderados determinados en el bloque 1004. Por ejemplo, de acuerdo con las ecuaciones (1) y (2) descritas anteriormente, un predictor final utilizado para descodificar un píxel de capa de mejora particular puede basarse en el producto del predictor de EL correspondiente al píxel y el peso asociado al predictor de EL, y también en el producto del predictor de BL correspondiente al píxel y el peso asociado al predictor de BL. Los predictores ponderados respectivos pueden sumarse para proporcionar un predictor final para el píxel particular.

[0126] La FIG. 11 ilustra un proceso para asignar pesos a los predictores de EL y BL en función de la ubicación de píxel con un bloque de EL o en relación con bloques de EL vecinos codificados previamente. El proceso puede

implementarse en uno o más de los dispositivos descritos en el presente documento, tal como el dispositivo de origen 12 del codificador de vídeo 20 o el descodificador de vídeo del dispositivo de destino 14.

5 **[0127]** El proceso comienza en el bloque 1102, donde se determina la ubicación de un píxel particular en el EL. En particular, la distancia desde esa ubicación a un píxel en un bloque de EL previamente codificado (por ejemplo, un bloque de EL que es adyacente al bloque de EL que está siendo codificado), en la dirección de predicción utilizada para el bloque de EL actual, puede ser inversamente proporcional al peso dado al predictor de EL.

10 **[0128]** En el bloque 1104, el dispositivo que ejecuta el proceso puede asignar un peso al predictor de EL para el píxel actual (o fila o columna) en función de la ubicación y la distancia de los bloques de EL codificados previamente, como se determinó anteriormente. De manera ilustrativa, el peso asignado al predictor de EL puede ser cercano o igual a 1, donde el píxel actual es un vecino directo al píxel codificado previamente en la dirección de predicción. El predictor de EL puede reducirse o acercarse a cero a medida que aumenta la distancia entre el píxel actual y el píxel codificado previamente en la dirección de predicción.

15 **[0129]** En el bloque 1106, el dispositivo que ejecuta el proceso puede asignar un peso al predictor de BL basándose en el peso asignado al predictor de EL. De manera ilustrativa, los pesos asignados al predictor de EL y al predictor de BL para cualquier píxel pueden sumarse a una constante, como 1. En algunos modos de realización, el peso del predictor de BL se asigna en función de la ubicación del píxel de EL actual, y el peso asignado al predictor de EL se basa en el peso asignado al predictor de BL.

[0130] En el bloque 1108, el dispositivo que ejecuta el proceso puede repetir el proceso de determinar la ubicación de píxel y asignar pesos en función de esa ubicación para cada píxel en el bloque de EL actual.

25 **[0131]** La FIG. 12 ilustra un proceso para determinar o asignar pesos de predictor de BL y EL según las características de la capa base y la capa de mejora. El proceso puede implementarse en uno o más de los dispositivos descritos en el presente documento, tal como el dispositivo de origen 12 o el dispositivo de destino 14.

30 **[0132]** El proceso comienza en el bloque 1202, donde se determina la característica pertinente de la capa base y la capa de mejora. La característica puede ser el parámetro de cuantificación, el factor de escala o alguna otra característica pertinente para los predictores de ponderación.

35 **[0133]** En el bloque 1204, el peso del predictor de EL se determina o asigna en función de las características determinadas anteriormente. Por ejemplo, se puede asignar un peso de predictor de EL relativamente grande si la diferencia entre los parámetros de cuantificación o los factores de escalamiento de la capa de mejora y la capa base es grande. Una diferencia tan grande puede indicar que bloques de EL previamente codificados ofrecen mejores predictores para el bloque de EL actual que un bloque de BL porque, por ejemplo, la calidad del bloque de BL puede ser significativamente menor que la calidad de los bloques de EL.

40 **[0134]** En el bloque 1206, el dispositivo que ejecuta el proceso puede asignar un peso al predictor de BL basándose en el peso asignado al predictor de EL. De manera ilustrativa, los pesos asignados al predictor de EL y al predictor de BL pueden sumarse a una constante, como 1. En algunos modos de realización, el peso del predictor de BL se asigna en función de las características del bloque de EL y del bloque de BL, y el peso asignado al predictor de EL está basado en el peso asignado al predictor de BL.

45 **[0135]** En uno o más ejemplos, las funciones descritas pueden implementarse en hardware, software, firmware o cualquier combinación de los mismos. Si se implementan en software, las funciones se pueden almacenar en, o transmitir por, un medio legible por ordenador, como una o más instrucciones o código, y ejecutarse mediante una unidad de procesamiento basada en hardware. Los medios legibles por ordenador pueden incluir medios de almacenamiento legibles por ordenador, que corresponden a un medio tangible tales como medios de almacenamiento de datos, o medios de comunicación que incluyen cualquier medio que facilite la transferencia de un programa informático desde un lugar a otro, por ejemplo, de acuerdo con un protocolo de comunicación. De esta manera, los medios legibles por ordenador pueden corresponder, en general, a (1) medios de almacenamiento tangibles legibles por ordenador que sean no transitorios o (2) un medio de comunicación tal como una señal o una onda portadora. Los medios de almacenamiento de datos pueden ser cualquier medio disponible al que se pueda acceder desde uno o más ordenadores o uno o más procesadores para recuperar instrucciones, código y/o estructuras de datos para la implementación de las técnicas descritas en esta divulgación. Un producto de programa informático puede incluir un medio legible por ordenador.

60 **[0136]** A modo de ejemplo, y no de limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento de disco óptico, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, memoria flash o cualquier otro medio que pueda usarse para almacenar código de programa deseado en forma de instrucciones o estructuras de datos y al que pueda accederse mediante un ordenador. Además, cualquier conexión recibe debidamente la denominación de medio legible por ordenador. Por ejemplo, si las instrucciones se transmiten desde un sitio web, un servidor u otro origen remoto usando un cable coaxial, un cable de fibra óptica, un par trenzado, una línea de abonado digital

65

(DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, entonces el cable coaxial, el cable de fibra óptica, el par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Sin embargo, debería entenderse que los medios de almacenamiento legibles por ordenador y los medios de almacenamiento de datos no incluyen conexiones, ondas portadoras, señales u otros medios transitorios, sino que, en cambio, se orientan a medios de almacenamiento tangibles no transitorios. El término disco, como se usa en el presente documento, incluye el disco compacto (CD), el disco láser, el disco óptico, el disco versátil digital (DVD), el disco flexible y el disco Blu-ray, donde algunos discos reproducen normalmente datos de manera magnética, mientras que otros discos reproducen datos de manera óptica con láseres. Las combinaciones de lo anterior también deberían incluirse dentro del alcance de los medios legibles por ordenador.

[0137] Las instrucciones pueden ser ejecutadas por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados específicos de la aplicación (ASIC), formaciones de compuertas programables in situ (FPGA) u otros circuitos lógicos, integrados o discretos, equivalentes. En consecuencia, el término "procesador", como se usa en el presente documento, se puede referir a cualquiera de las estructuras anteriores o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento se puede proporcionar dentro de módulos de hardware y/o software dedicados configurados para la codificación y la descodificación, o incorporarse en un códec combinado. Además, las técnicas se podrían implementar totalmente en uno o más circuitos o elementos lógicos.

[0138] Las técnicas de la esta divulgación se pueden implementar en una amplia variedad de dispositivos o aparatos, incluidos un equipo manual inalámbrico, un circuito integrado (IC) o un conjunto de IC (por ejemplo, un conjunto de chips). Diversos componentes, módulos o unidades se describen en esta divulgación para enfatizar aspectos funcionales de dispositivos configurados para realizar las técnicas divulgadas, pero no requieren necesariamente su realización mediante diferentes unidades de hardware. En cambio, como se ha descrito anteriormente, diversas unidades se pueden combinar en una unidad de hardware de códec, o ser proporcionadas por un grupo de unidades de hardware interoperativas, incluidos uno o más procesadores descritos anteriormente, junto con software y/o firmware adecuados.

[0139] Se han descrito diversos ejemplos. Estos y otros ejemplos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un procedimiento de codificación de vídeo digital, comprendiendo el procedimiento:

5 obtener información de vídeo asociada a una capa base y una capa de mejora correspondiente;
determinar un primer peso asociado a un predictor de capa base, P_b , y un segundo peso asociado a un predictor de capa de mejora, P_e , en el que:

10 el predictor de capa base, P_b , se forma utilizando la textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente (700), y el predictor de capa de mejora, P_e , se forma usando la intrapredicción de un bloque de capa de mejora a partir de un bloque vecino de píxeles reconstruidos;

15 codificar un píxel actual en la trama de la capa de mejora en función de un predictor final, P_f , en el que el predictor final, P_f , comprende una combinación del predictor de capa base, P_b , ponderado de acuerdo con el primer peso y del predictor de capa de mejora, P_e , ponderado de acuerdo con el segundo peso, **caracterizado por que:**

20 el primer peso y el segundo peso se determinan en función de una distancia (1102) entre la ubicación de píxel actual codificada en un bloque de capa de mejora actual (700) y un borde (702, 706) de un bloque de capa de mejora vecino (720), y en el que:

25 el segundo peso aumenta, en general, para los píxeles más cercanos al borde (710);
el primer peso aumenta, en general, para los píxeles más alejados del borde (1106); y
en cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada de 1 (1106).

30 2. Un procedimiento de descodificación de vídeo digital, comprendiendo el procedimiento:

obtener información de vídeo asociada a una capa base y una capa de mejora correspondiente;
35 determinar un primer peso asociado a un predictor de capa base, P_b , y un segundo peso asociado a un predictor de capa de mejora, P_e , en el que:

40 el predictor de capa base, P_b , se forma utilizando la textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente (700), y el predictor de capa de mejora, P_e , se forma usando la intrapredicción de un bloque de capa de mejora a partir de un bloque vecino de píxeles reconstruidos;

45 descodificar un píxel actual en la trama de la capa de mejora en función de un predictor final, P_f , en el que el predictor final, P_f , comprende una combinación del predictor de capa base, P_b , ponderado de acuerdo con el primer peso y del predictor de capa de mejora, P_e , ponderado de acuerdo con el segundo peso, **caracterizado por que:**

50 el primer peso y el segundo peso se determinan en función de una distancia (1102) entre la ubicación de píxel actual codificada en un bloque de capa de mejora actual (700) y un borde (702, 706) de un bloque de capa de mejora vecino (720), y

en el que:

55 el segundo peso aumenta, en general, para los píxeles más cercanos al borde (710);
el primer peso aumenta, en general, para los píxeles más alejados del borde (1106); y para cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada de 1 (1106).

60 3. El procedimiento según la reivindicación 1, en el que el segundo peso es mayor que el primer peso para píxeles de capa de mejora sustancialmente adyacentes a bloques de capa de mejora codificados previamente.

65 4. El procedimiento según la reivindicación 2, en el que el segundo peso es mayor que el primer peso para píxeles de capa de mejora sustancialmente adyacentes a bloques de capa de mejora descodificados previamente.

5. El procedimiento según la reivindicación 1, en el que el primer peso es mayor que el segundo peso para los píxeles de capa de mejora más alejados de los bloques de capa de mejora codificados previamente.
- 5 6. El procedimiento según la reivindicación 2, en el que el primer peso es mayor que el segundo peso para los píxeles de capa de mejora más alejados de los bloques de capa de mejora descodificados previamente.
7. El procedimiento según la reivindicación 1 o la reivindicación 2, en el que la información de vídeo comprende uno o más de una dirección de intrapredicción, una ubicación de píxel, un parámetro de cuantificación y una relación de escalamiento espacial.
- 10
8. El procedimiento según la reivindicación 1 o la reivindicación 2, en el que el segundo peso se determina en función tanto de un parámetro de cuantificación de la capa base, QP1, como de un parámetro de cuantificación de la capa de mejora, QP2.
- 15
9. El procedimiento según la reivindicación 8, en el que el segundo peso aumenta, en general, a medida que aumenta el valor de QP1 - QP2.
10. El procedimiento según la reivindicación 1 o la reivindicación 2, en el que el segundo peso se determina en función de la relación de escalamiento espacial.
- 20
11. El procedimiento según la reivindicación 10, en el que el segundo peso aumenta, en general, a medida que aumenta la relación de escalamiento espacial.
- 25
12. Un medio de almacenamiento legible por ordenador, que comprende instrucciones ejecutables por un procesador de un aparato, donde las instrucciones hacen que el aparato realice el procedimiento de una cualquiera de las reivindicaciones anteriores.
- 30
13. Un aparato para codificar vídeo digital, comprendiendo el aparato:
- una memoria configurada para almacenar información de vídeo asociada a una capa base y una capa de mejora correspondiente; y
- un procesador en comunicación con la memoria, estando configurado el procesador para:
- 35
- determinar un primer peso asociado a un predictor de capa base, P_b , y un segundo peso asociado a un predictor de capa de mejora, P_e , en el que el predictor de capa base, P_b , se forma usando la textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente (700), y el predictor de capa de mejora, P_e , se forma utilizando la intrapredicción de un bloque de capa de mejora a partir de un bloque vecino de píxeles reconstruidos;
- 40
- y
- codificar un píxel actual en la trama de la capa de mejora en función de un predictor final, P_f , en el que el predictor final, P_f , comprende una combinación del predictor de capa base, P_b , ponderado de acuerdo con el primer peso y del predictor de capa de mejora, P_e , ponderado de acuerdo con el segundo peso, **caracterizado por que:**
- 45
- el primer peso y el segundo peso se determinan en función de una distancia (1102) entre la ubicación de píxel actual codificada en un bloque de capa de mejora actual (700) y un borde (702, 706) de un bloque de capa de mejora vecino (720), y
- 50
- en el que:
- el segundo peso aumenta, en general, para los píxeles más cercanos al borde (710);
- 55
- el primer peso aumenta, en general, para los píxeles más alejados del borde (1106); y
- en cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada de 1 (1106).
- 60
14. Un aparato para descodificar vídeo digital, comprendiendo el aparato:
- una memoria configurada para almacenar información de vídeo asociada a una capa base y una capa de mejora correspondiente; y
- 65
- un procesador en comunicación con la memoria, estando configurado el procesador para:

5 determinar un primer peso asociado a un predictor de capa base, P_b , y un segundo peso asociado a un predictor de capa de mejora, P_e , en el que el predictor de capa base, P_b , se forma usando la textura reconstruida a partir de un bloque de capa base para predecir la textura de un bloque de capa de mejora correspondiente (700), y el predictor de capa de mejora, P_e , se forma utilizando la intrapredicción de un bloque de capa de mejora a partir de un bloque vecino de píxeles reconstruidos; y

10 decodificar un píxel actual en la trama de la capa de mejora en función de un predictor final, P_f , en el que el predictor final, P_f , comprende una combinación del predictor de capa base, P_b , ponderado de acuerdo con el primer peso y del predictor de capa de mejora, P_e , ponderado de acuerdo con el segundo peso, **caracterizado por que:**

15 el primer peso y el segundo peso se determinan en función de una distancia (1102) entre la ubicación de píxel actual codificada en un bloque de capa de mejora actual (700) y un borde (702, 706) de un bloque de capa de mejora vecino (720), y

en el que:

20 el segundo peso aumenta, en general, para los píxeles más cercanos al borde (710);

el primer peso aumenta, en general, para los píxeles más alejados del borde (1106); y

25 en cada ubicación de píxel, el primer peso y el segundo peso se suman a una constante predeterminada de 1 (1106).

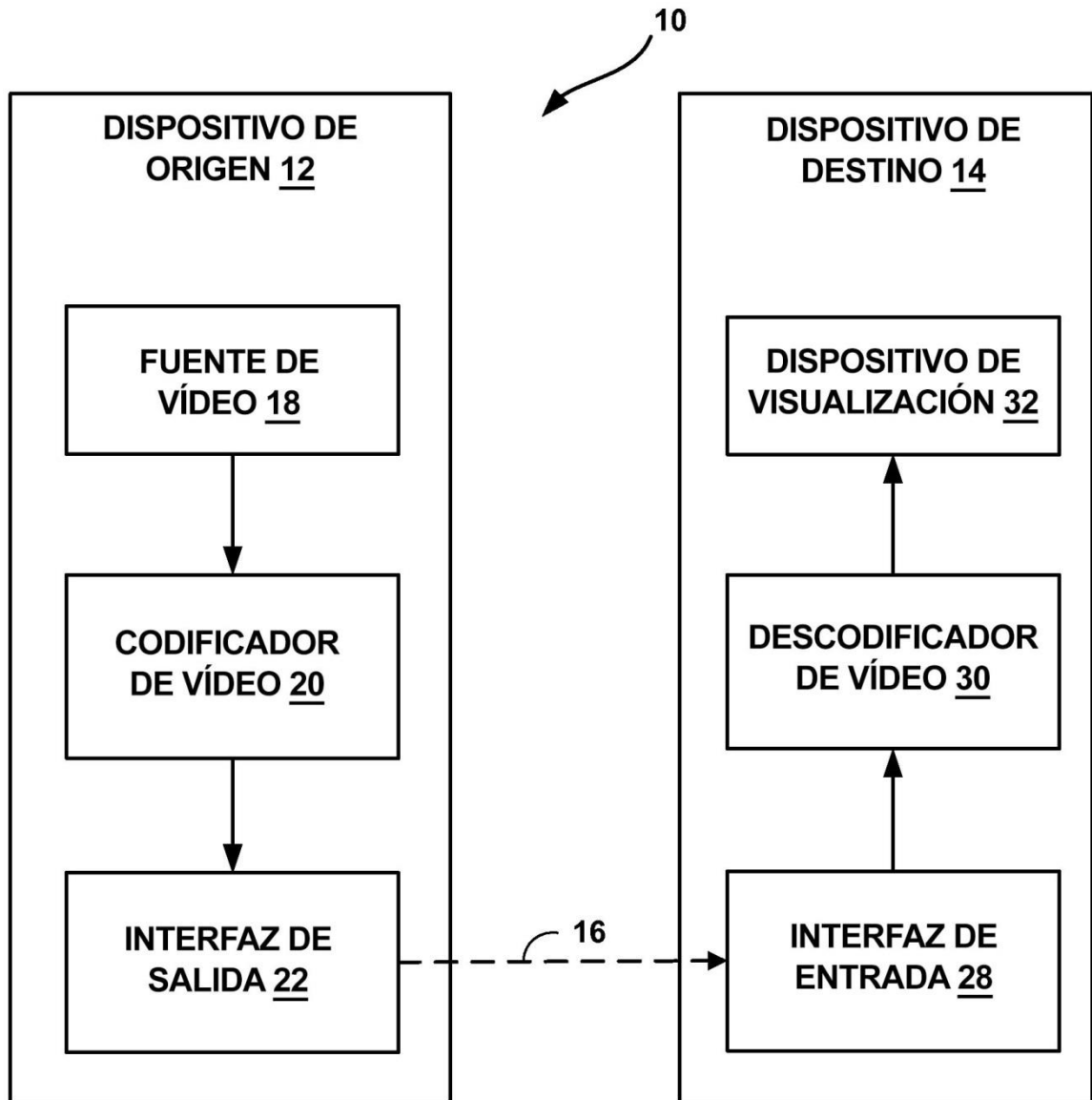


FIG. 1

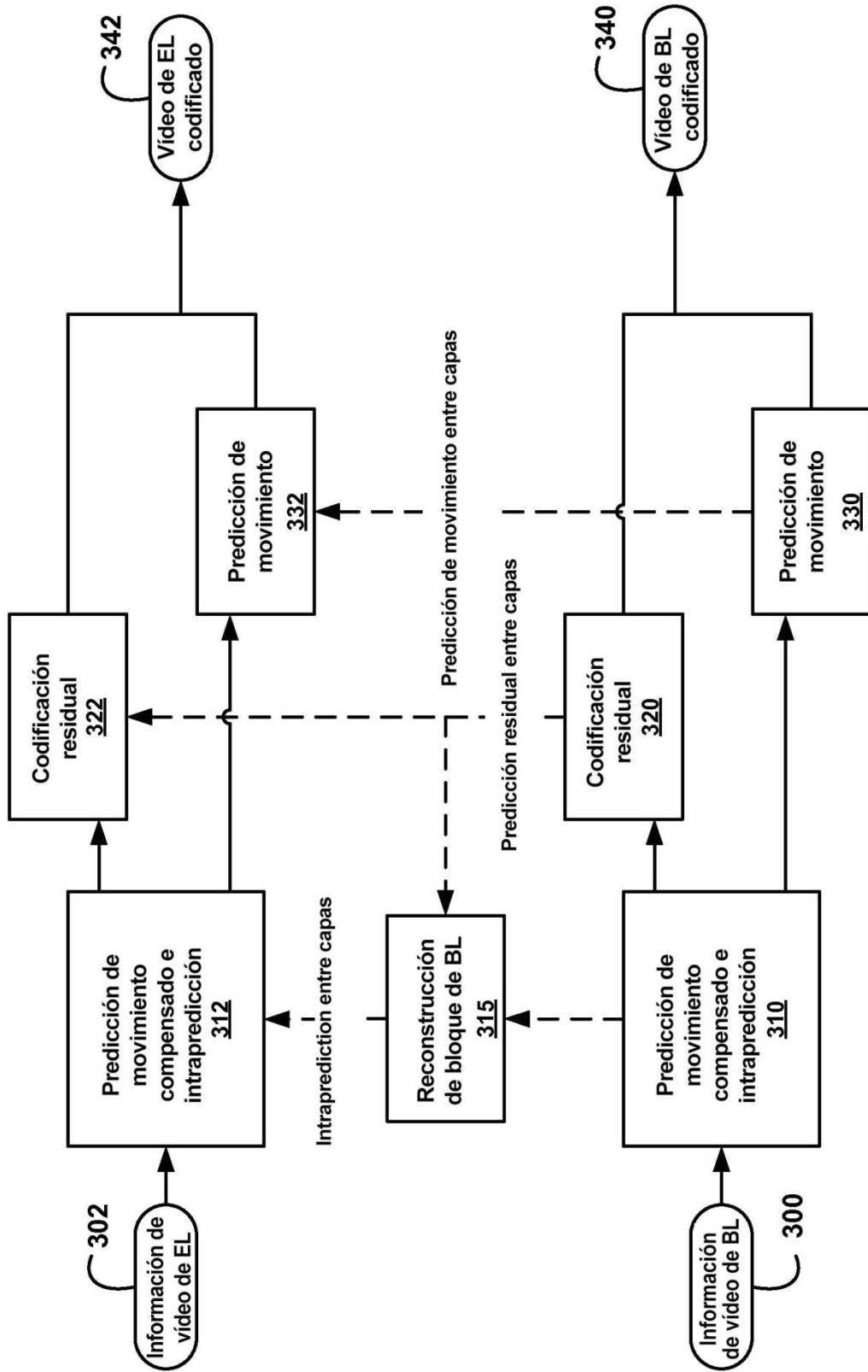


FIG. 2

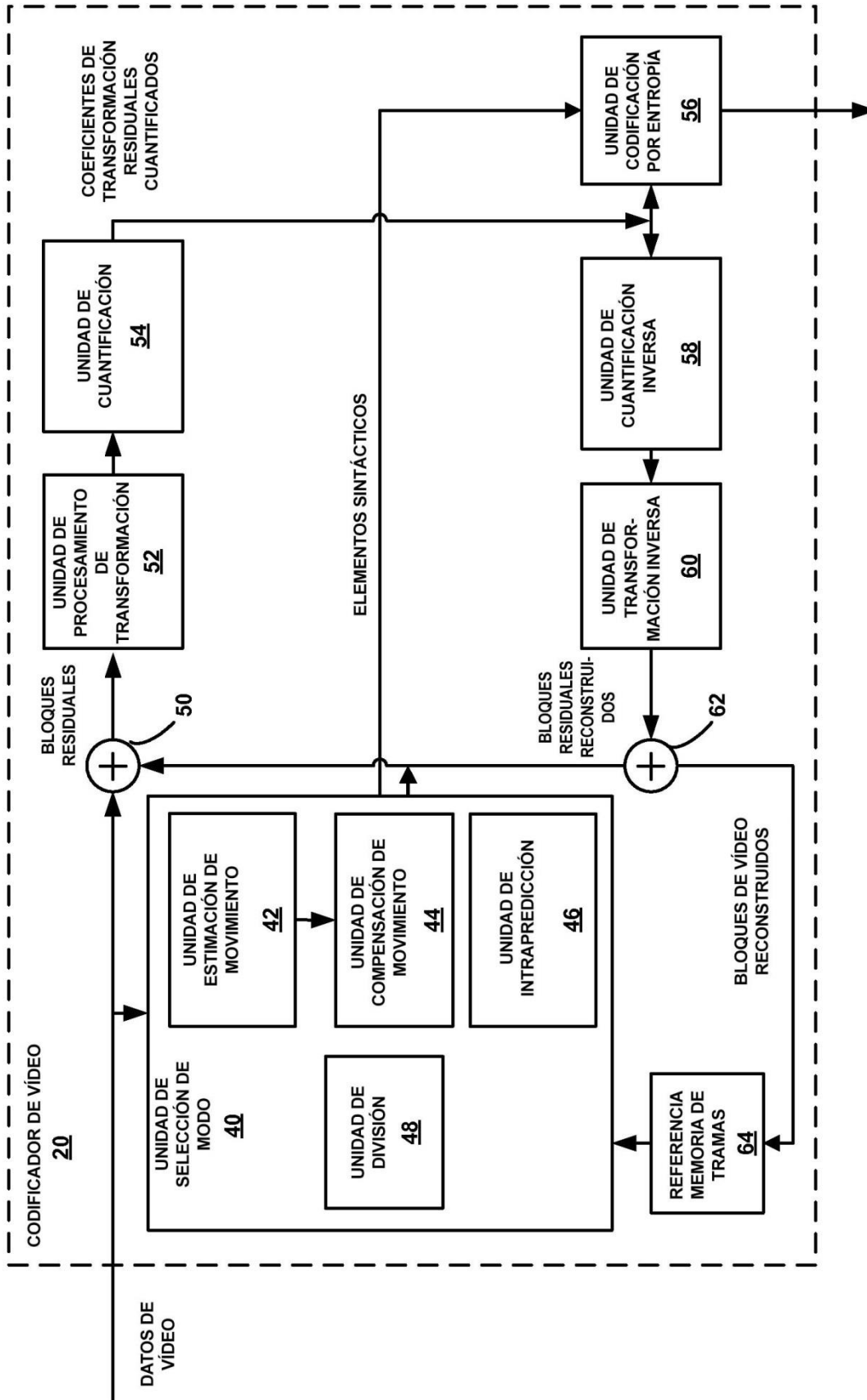


FIG. 3

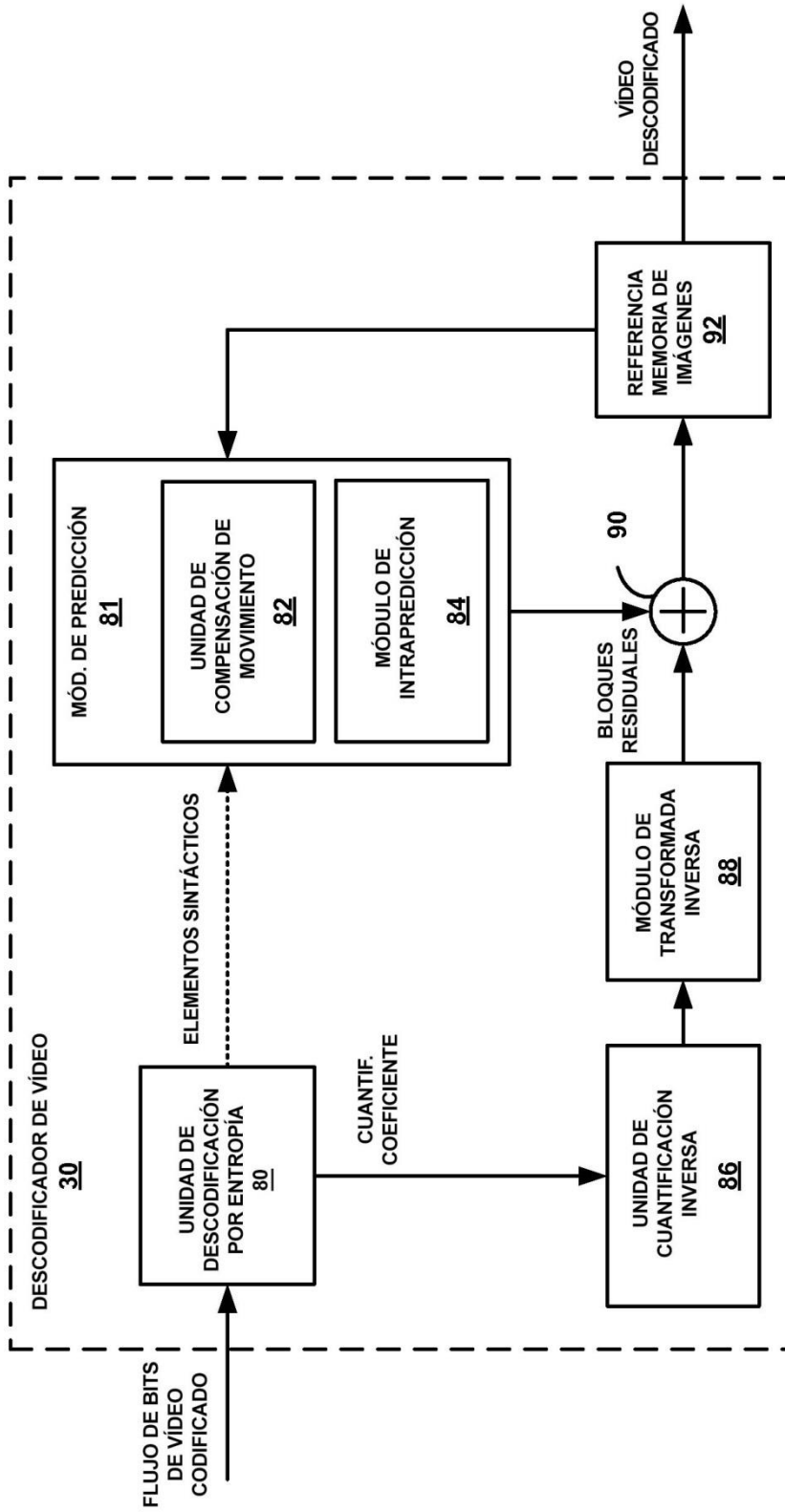


FIG. 4

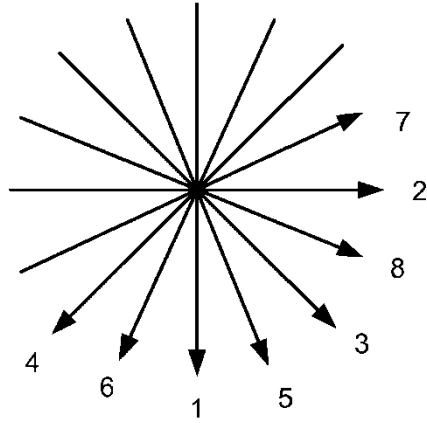


FIG. 5

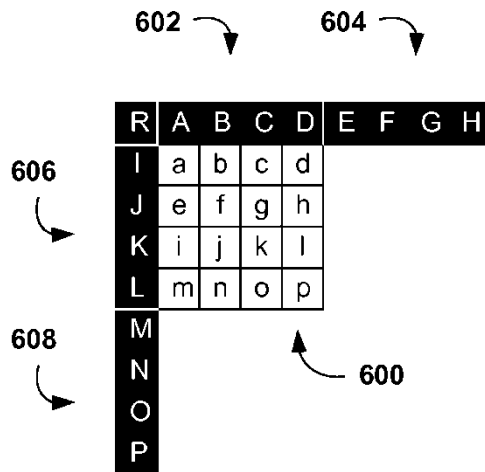


FIG. 6

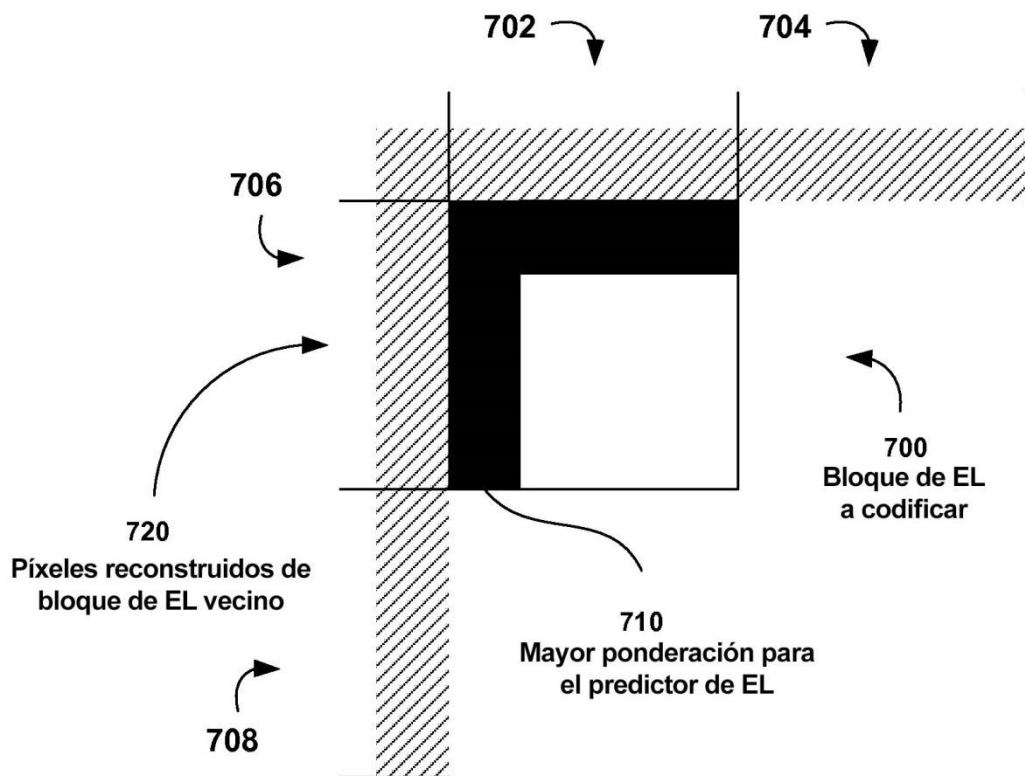


FIG. 7

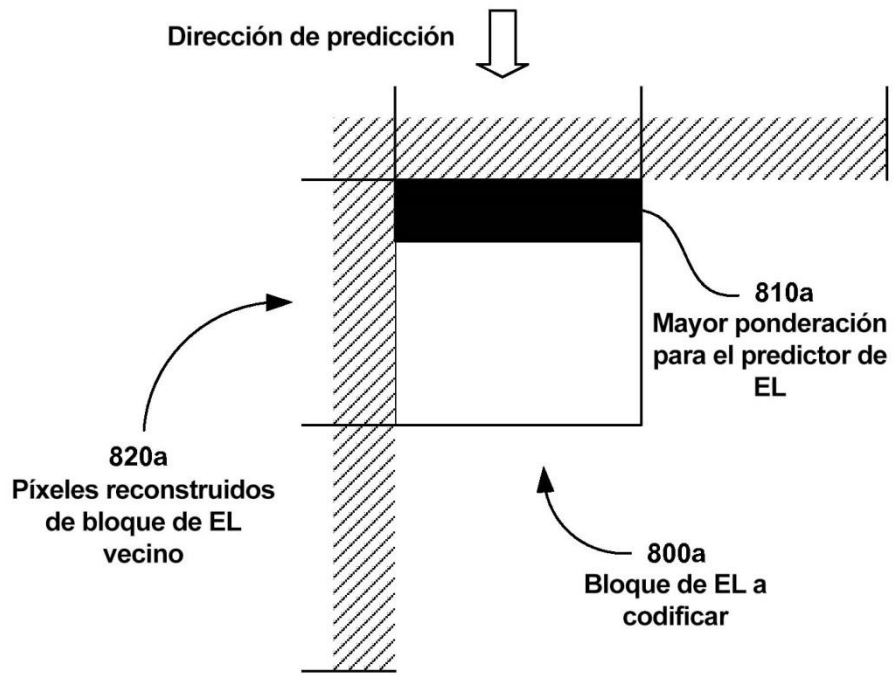


FIG. 8A

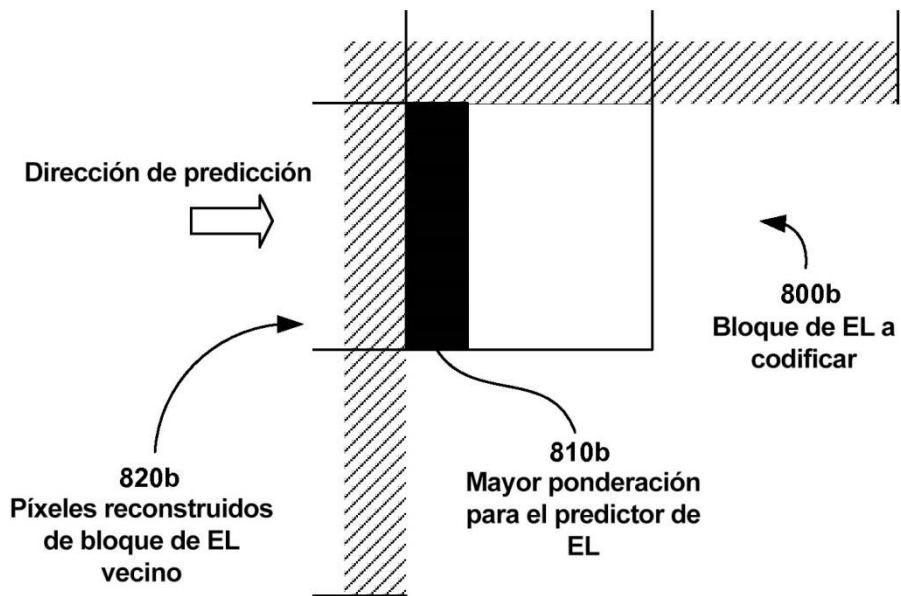


FIG. 8B

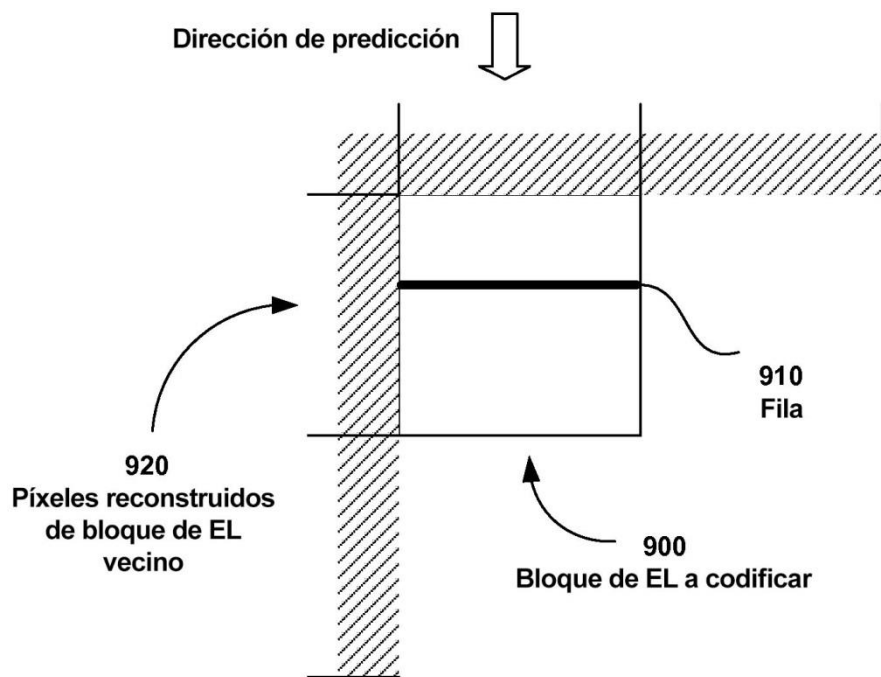


FIG. 9

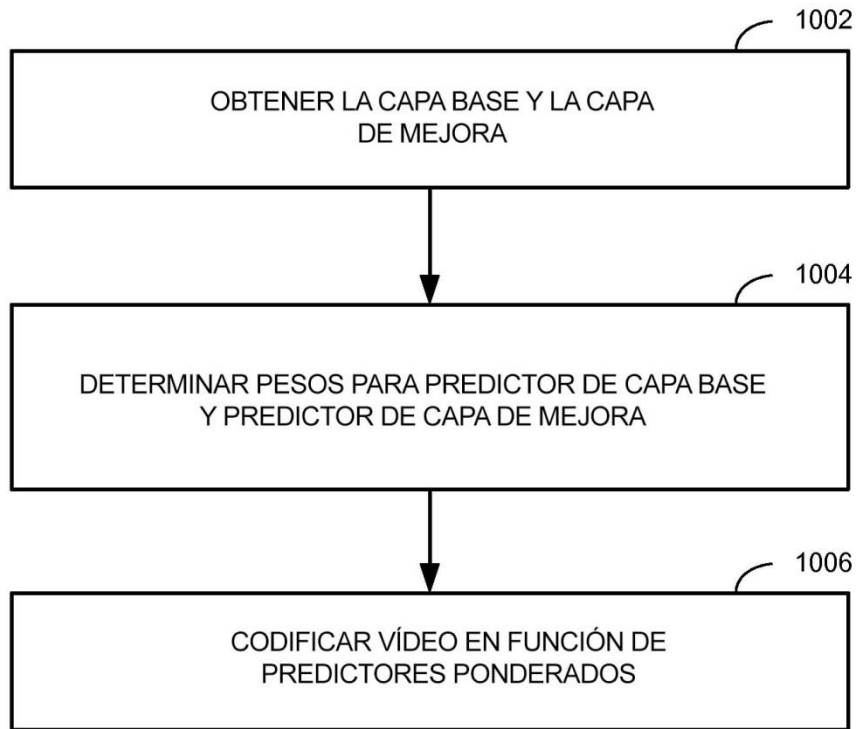


FIG. 10

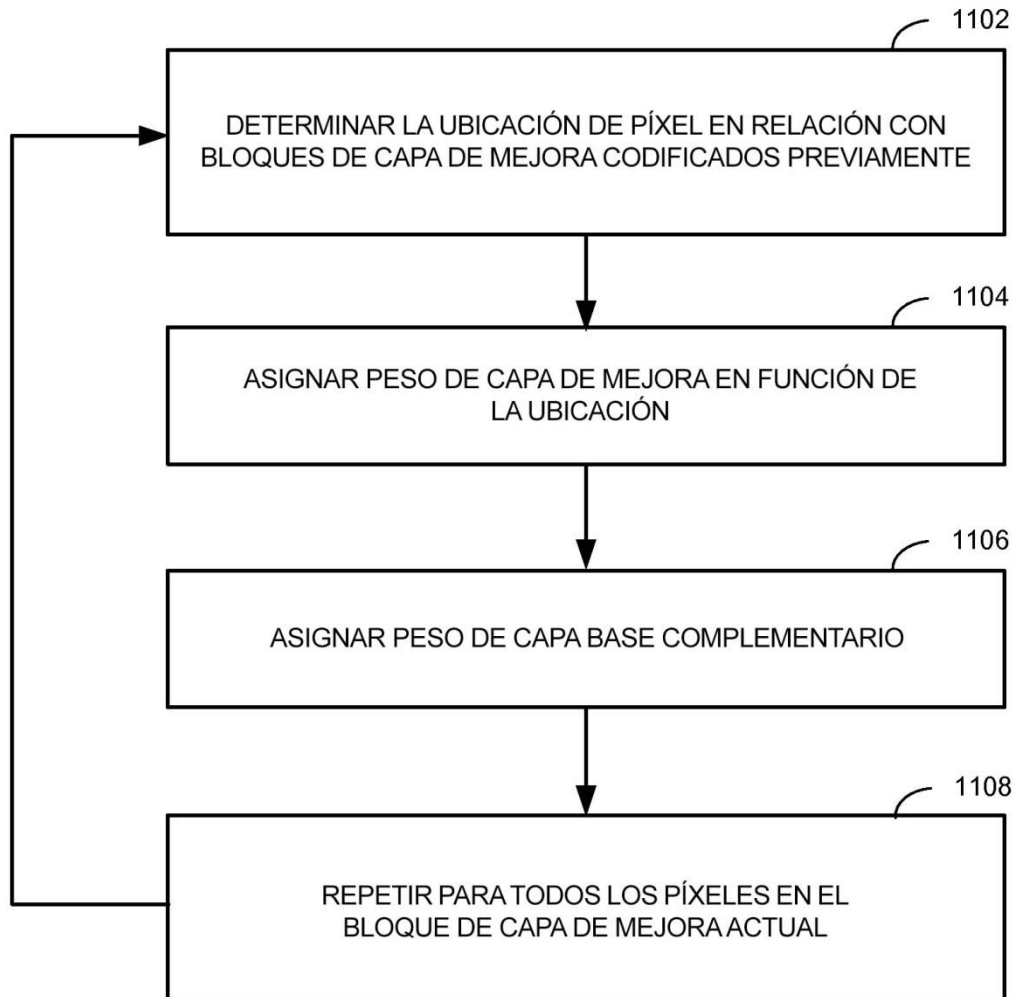


FIG. 11

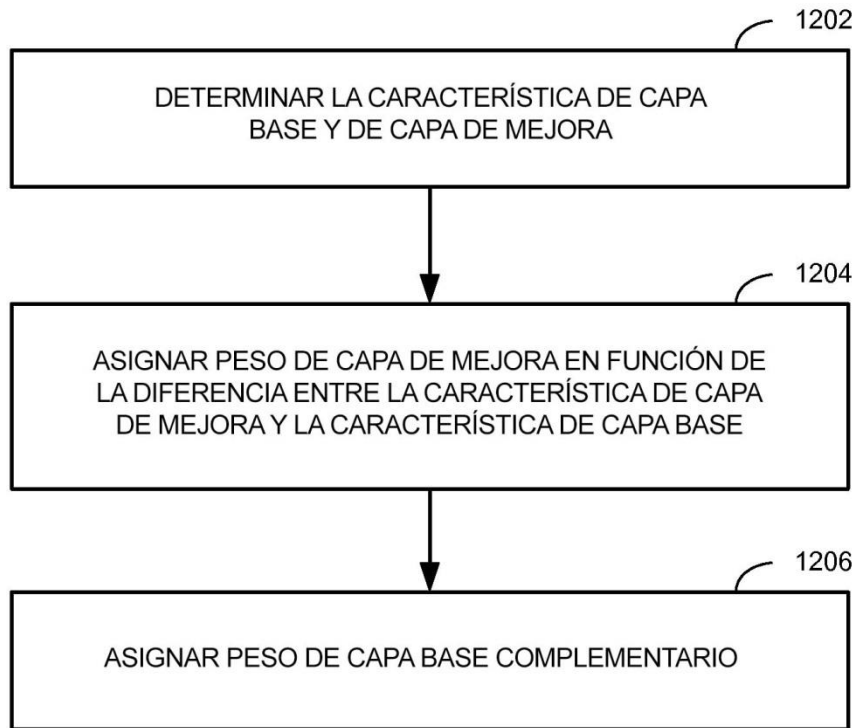


FIG. 12