

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 633 476**

51 Int. Cl.:

H04N 19/46 (2014.01)

H04N 19/117 (2014.01)

H04N 19/182 (2014.01)

H04N 19/82 (2014.01)

H04N 19/523 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **10.04.2009 E 14184120 (5)**

97 Fecha y número de publicación de la concesión europea: **17.05.2017 EP 2827593**

54 Título: **Filtrado de tipo interpolación de posiciones de píxeles enteros en la codificación de vídeo**

30 Prioridad:

10.04.2008 US 44023 P

10.04.2008 US 44020 P

11.04.2008 US 44240 P

30.05.2008 US 57373 P

08.04.2009 US 420292

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
21.09.2017

73 Titular/es:

QUALCOMM INCORPORATED (100.0%)

5775 Morehouse Drive

San Diego, CA 92121, US

72 Inventor/es:

YE, YAN y

KARCZEWICZ, MARTA

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 633 476 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

Filtrado de tipo interpolación de posiciones de píxeles enteros en la codificación de vídeo

5 CAMPO TÉCNICO

Esta divulgación se refiere a la codificación y decodificación de vídeo digital y, más particularmente, a técnicas de filtrado aplicadas para generar datos predictivos utilizados en la codificación y decodificación de vídeo.

10 ANTECEDENTES

Las capacidades del vídeo digital pueden incorporarse en una amplia gama de dispositivos, incluyendo televisiones digitales, sistemas de radiodifusión directa digitales, sistemas de radiodifusión inalámbricos, asistentes digitales personales (PDA), ordenadores portátiles o de escritorio, cámaras digitales, dispositivos de grabación digitales, dispositivos de videojuegos, consolas de videojuegos, teléfonos celulares o de radio por satélite, y similares. Los dispositivos de vídeo digital implementan técnicas de compresión de vídeo, tales como las descritas en estándares definidos por las normas MPEG-2, MPEG-4 o ITU-T H.264/MPEG-4, parte 10, Codificación avanzada de vídeo (AVC), para transmitir y recibir información de vídeo digital de manera más eficiente. Las técnicas de compresión de vídeo pueden realizar una predicción espacial y/o una predicción temporal para reducir o eliminar la redundancia inherente a las secuencias de vídeo.

La intercodificación basada en bloques es una técnica de codificación muy útil que se basa en la predicción temporal para reducir o eliminar la redundancia temporal entre bloques de vídeo de unidades codificadas sucesivas de una secuencia de vídeo. Las unidades codificadas pueden comprender tramas de vídeo, fragmentos de tramas de vídeo, grupos de imágenes u otra unidad definida de bloques de vídeo codificados. Para la intercodificación, un codificador de vídeo realiza estimación de movimiento y compensación de movimiento para rastrear el movimiento de los bloques de vídeo correspondientes de dos o más unidades codificadas adyacentes. La estimación de movimiento genera vectores de movimiento, que indican un desplazamiento de bloques de vídeo en relación con los correspondientes bloques de vídeo de predicción de una o más tramas de referencia u otras unidades codificadas. La compensación de movimiento utiliza los vectores de movimiento para generar bloques de vídeo de predicción a partir de la una o más tramas de referencia u otras unidades codificadas. Después de la compensación de movimiento, los bloques de vídeo residuales se forman restando bloques de vídeo de predicción de los bloques de vídeo originales que se están codificando.

El codificador de vídeo también puede aplicar procedimientos de codificación de transformada, cuantificación y entropía para reducir aún más la velocidad de bits asociada con la transmisión de bloques residuales. Las técnicas de transformada pueden comprender transformadas de coseno discretas (DCT) o procesos conceptualmente similares. De forma alternativa, pueden utilizarse transformadas de ondícula, transformadas enteras u otros tipos de transformadas. A título de ejemplo, en un proceso de DCT, un conjunto de valores de píxel se convierte en coeficientes de transformada, que puede representar la energía de los valores de píxel en el dominio de la frecuencia. La cuantificación se aplica a los coeficientes de transformada, y generalmente implica un proceso que reduce el número de bits asociados con cualquier coeficiente de transformada dado. La codificación por entropía comprende uno o más procesos que comprimen colectivamente una secuencia de modos de codificación, información de movimiento, patrones de bloques codificados y coeficientes de transformada cuantificados. Entre los ejemplos de codificación por entropía se incluyen, pero no se limitan a, la codificación de longitud variable adaptativa al contenido (CAVLC) y la codificación aritmética binaria adaptativa al contexto (CABAC).

Un bloque de vídeo codificado puede representarse mediante información de predicción que se puede utilizar para crear o identificar un bloque predictivo, y un bloque residual de datos indicativos de diferencias entre el bloque que se está codificando y el bloque predictivo. La información de predicción puede comprender uno o más vectores de movimiento que se utilizan para identificar el bloque de datos predictivo. Dados los vectores de movimiento, el decodificador puede ser capaz de reconstruir los bloques predictivos que se utilizaron para codificar el residuo. Por lo tanto, dado un conjunto de bloques residuales y un conjunto de vectores de movimiento (y posiblemente sintaxis adicional), el decodificador puede reconstruir una trama de vídeo que se codificó originalmente. La intercodificación basada en estimación de movimiento y compensación de movimiento puede alcanzar muy buena compresión, porque las tramas de vídeo sucesivas u otros tipos de unidades codificadas a menudo son muy similares. Una secuencia de vídeo codificada puede comprender bloques de datos residuales, vectores de movimiento y posiblemente otros tipos de sintaxis.

Se han desarrollado técnicas de interpolación para mejorar el nivel de compresión que puede lograrse en la intercodificación. En este caso, los datos predictivos generados durante la compensación de movimiento, que se usan para codificar un bloque de vídeo, pueden interpolarse a partir de los píxeles de bloques de vídeo de la trama de vídeo u otra unidad codificada utilizada en la estimación de movimiento. La interpolación se realiza a menudo para generar valores de medio píxel predictivos (medio pel) y valores de cuarto de píxel predictivos (cuarto de pel). Los valores de medio y de cuarto pel están asociados con ubicaciones de subpíxeles. Pueden usarse vectores de movimiento fraccional para identificar bloques de vídeo en la resolución de subpíxeles con el fin de capturar el

movimiento fraccional en una secuencia de vídeo y, de este modo, proporcionar bloques predictivos que son más similares a los bloques de vídeo que se están codificando los bloques de vídeo de enteros. El documento EP1641274 A1 describe un enfoque para reducir la transmisión de errores de predicción eliminando el grano de película de aquellas imágenes que se usan como imágenes de referencia para la codificación predictiva. Esto se consigue incluyendo un filtro de separación junto con un filtro de desbloqueo en el bucle de decodificación.

SUMARIO

En general, esta divulgación describe técnicas de filtrado aplicadas por un codificador y un decodificador durante la fase de predicción de un proceso de codificación y/o decodificación de vídeo. Las técnicas de filtrado descritas pueden mejorar la precisión de los datos predictivos utilizados durante la interpolación fraccional y, en algunos casos, pueden mejorar los datos predictivos de bloques de enteros de píxeles. Hay varios aspectos de esta descripción, incluyendo un soporte de filtro de doce píxeles útil que se puede usar para interpolación, técnicas que utilizan simetría de coeficiente y simetría de píxeles para reducir la cantidad de datos que se necesitan enviar entre un codificador y un decodificador para configurar el soporte de filtro para interpolación y técnicas para filtrar datos en ubicaciones de píxeles enteros de una manera similar a la interpolación de subpíxeles. Otros aspectos de esta divulgación se refieren a técnicas para codificar información en el flujo de datos para transportar el tipo de filtro utilizado y, posiblemente, los coeficientes de filtro utilizados. También se describen técnicas de codificación predictiva para coeficientes de filtro. Estos y otros aspectos de esta divulgación resultarán evidentes a partir de la siguiente descripción.

En un ejemplo, esta divulgación proporciona un método que comprende obtener bloques de píxeles, en el que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles, filtrar los valores de píxeles enteros basados en otros valores de píxeles enteros dentro del bloque de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y generar un bloque de predicción basado en los valores de píxeles enteros ajustados.

El filtrado de píxeles enteros dentro de un proceso de compensación de movimiento se conoce a partir del filtro de desbloqueo AVC/H.264 como se divulga en "Adaptive deblocking filter", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, vol. 13, n.º 7, 1 de julio de 2003.

Un concepto de filtrado de píxeles enteros en un bucle de predicción para reducir el ruido de una imagen de entrada también se conoce a partir del documento EP1641274.

En otro ejemplo, esta divulgación proporciona un aparato que comprende un codificador de vídeo que obtiene bloques de píxeles, en el que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles, filtra los valores de píxeles enteros basados en otros valores de píxeles enteros dentro del bloque de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y genera un bloque de predicción basado en los valores de píxeles enteros ajustados.

En otro ejemplo, esta divulgación proporciona un dispositivo que comprende medios para obtener bloques de píxeles, en el que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles, medios para filtrar los valores de píxeles enteros basados en otros valores de píxeles enteros dentro del bloque de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y medios para generar un bloque de predicción basado en los valores de píxeles enteros ajustados.

Las técnicas descritas en la presente divulgación pueden implementarse en hardware, software, firmware o cualquier combinación de estos. Si se implementa en software, el software puede ejecutarse en uno o más procesadores, tales como un microprocesador, un circuito integrado específico de la aplicación (ASIC), una matriz de puertas programables in situ (FPGA) o un procesador de señales digitales (DSP). El software que ejecuta las técnicas se puede almacenar inicialmente en un medio legible por ordenador y cargar y ejecutar en el procesador.

Por consiguiente, esta divulgación también contempla un medio de almacenamiento legible por ordenador que comprende instrucciones que, al ejecutarse por un procesador, hacen que el procesador obtenga bloques de píxeles, en el que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles, filtre los valores de píxeles enteros basados en otros valores de píxeles enteros dentro del bloque de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y genere un bloque de predicción basado en los valores de píxeles enteros ajustados.

Los detalles de uno o más aspectos de la divulgación se exponen en los dibujos adjuntos y la descripción siguiente. Otras características, objetivos y ventajas de las técnicas descritas en la presente divulgación resultarán evidentes a partir de la descripción y los dibujos, y a partir de las reivindicaciones.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

- 5 La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo ejemplar que puede implementar las técnicas de esta divulgación.
- La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo que puede realizar las técnicas de filtrado coherentes con esta divulgación.
- 10 La FIG. 3 es un diagrama conceptual que ilustra posiciones de píxeles enteros asociadas con datos de predicción, y posiciones de subpíxeles asociadas con datos de predicción interpolados.
- La FIG. 4 es un diagrama conceptual que ilustra un soporte de filtro de 12 píxeles con respecto a nueve ubicaciones de subpíxeles.
- 15 La FIG. 5 es un diagrama conceptual que ilustra un soporte de filtro horizontal de 6 píxeles con respecto a tres ubicaciones de subpíxeles horizontales, y un soporte de filtro vertical de 6 píxeles con respecto a tres ubicaciones verticales de subpíxeles.
- 20 La FIG. 6 es un diagrama conceptual que ilustra un soporte de filtro de cinco píxeles por cinco píxeles para filtrar una ubicación de píxeles enteros.
- La FIG. 7 es un diagrama conceptual que ilustra cuatro posiciones de píxeles enteros y quince posiciones de subpíxeles con sombreado para posiciones de píxeles en grupos que pueden utilizar simetría de píxeles para coeficientes de filtro coherentes con esta divulgación.
- 25 La FIG. 8 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales horizontales con relación a un subpíxel, con sombreado que muestra simetría de coeficientes.
- 30 La FIG. 9 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales horizontales con relación a un subpíxel, con sombreado que muestra una falta de simetría de coeficientes.
- La FIG. 10 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales verticales con relación a un subpíxel, con sombreado que muestra simetría de coeficientes.
- 35 La FIG. 11 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales verticales con relación a un subpíxel, con sombreado que muestra una falta de simetría de coeficientes.
- La FIG. 12 es un diagrama conceptual que ilustra dos posiciones de soporte de píxeles bidimensionales con relación a un subpíxel, con sombreado que muestra una falta de simetría de coeficientes.
- 40 La FIG. 13 es un diagrama conceptual que ilustra dos posiciones de soporte de píxeles bidimensionales con relación a un subpíxel, con sombreado que muestra simetría de coeficientes.
- 45 La FIG. 14 es un diagrama conceptual que ilustra dos posiciones de soporte de píxeles bidimensionales con relación a un subpíxel, con sombreado que muestra simetría de coeficientes.
- La FIG. 15 es un diagrama conceptual que ilustra dos posiciones de soporte de píxeles bidimensionales con relación a un subpíxel, con sombreado que muestra simetría de coeficientes.
- 50 La FIG. 16 es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo, que puede decodificar una secuencia de vídeo que se codifica de manera descrita en el presente documento.
- La FIG. 17 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza un soporte de filtro de doce píxeles coherente con esta divulgación.
- 55 La FIG. 18 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza un soporte de filtro de doce píxeles coherente con esta divulgación.
- La FIG. 19 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza simetría de coeficientes y simetría de píxeles coherente con esta divulgación.
- 60 La FIG. 20 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza simetría de coeficientes y simetría de píxeles coherente con esta divulgación.
- 65 La FIG. 21 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza el

filtrado de ubicaciones de píxeles enteros para generar valores de píxeles enteros ajustados consistentes con esta divulgación.

5 La FIG. 22 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza el filtrado de ubicaciones de píxeles enteros para generar valores de píxeles enteros ajustados consistentes con esta divulgación.

10 La FIG. 23 es un diagrama de flujo que ilustra una técnica para la interpolación definida por distorsión de velocidad para la codificación de vídeo basada en un filtro fijo o un filtro adaptativo.

La FIG. 24 es un diagrama de flujo que ilustra una técnica para codificar coeficientes de filtro usando codificación predictiva.

15 La FIG. 25 es otro diagrama de flujo que ilustra una técnica para codificar coeficientes de filtro usando codificación predictiva.

La FIG. 26 es un diagrama de flujo que ilustra una técnica para decodificar coeficientes de filtro usando codificación predictiva.

20 Las FIGS. 27 y 28 son gráficos conceptuales que ilustran coeficientes de filtro que pueden codificarse de forma predictiva.

La FIG. 29 es un ejemplo ilustrativo de una matriz de coeficientes de filtro de píxeles enteros para los que pueden usarse técnicas de predicción para la codificación.

25

DESCRIPCIÓN DETALLADA

30 Esta divulgación describe técnicas de filtrado aplicadas por un codificador y un decodificador durante la fase de predicción de un proceso de codificación y/o decodificación de vídeo. Las técnicas de filtrado descritas pueden mejorar la precisión de los datos predictivos utilizados durante la interpolación fraccional y, en algunos casos, pueden mejorar los datos predictivos de bloques de enteros de píxeles. Hay varios aspectos de esta descripción, incluyendo un soporte de filtro de doce píxeles útil que se puede usar para interpolación, técnicas que utilizan simetría de coeficiente y simetría de píxeles para reducir la cantidad de datos que se necesitan enviar entre un codificador y un decodificador para configurar el soporte de filtro para interpolación y técnicas para filtrar datos en
35 ubicaciones de píxeles enteros de una manera similar a la interpolación de subpíxeles. Estas y otras técnicas se describen en detalle a continuación.

40 La FIG. 1 es un diagrama de bloques que ilustra un sistema de codificación y decodificación de vídeo ejemplar 10 que puede usarse para implementar una o más de las técnicas de esta divulgación. Como se muestra en la FIG. 1, el sistema 10 incluye un dispositivo de origen 12 que transmite vídeo codificado a un dispositivo de destino 16 a través de un canal de comunicación 15. El dispositivo de origen 12 y el dispositivo de destino 16 pueden comprender cualquiera de una amplia gama de dispositivos. En algunos casos, el dispositivo de origen 12 y el dispositivo de destino 16 comprenden dispositivos de comunicación inalámbricos, tales como microteléfonos inalámbricos, los denominados radioteléfonos celulares o vía satélite, o cualquier dispositivo inalámbrico que pueda comunicar
45 información de vídeo a través de un canal de comunicación 15, donde el canal de comunicación 15 es inalámbrico en este caso. Sin embargo, las técnicas de esta divulgación, que se refieren al filtrado y la generación de datos predictivos durante la codificación predictiva, no están necesariamente limitadas a aplicaciones o configuraciones inalámbricas. Las técnicas también pueden ser útiles en una amplia gama de configuraciones y dispositivos diferentes, incluidos los dispositivos que se comunican a través de cables físicos, fibras ópticas u otros medios físicos o inalámbricos. Además, las técnicas de codificación o decodificación también pueden aplicarse en un dispositivo autónomo que no necesariamente se comunica con otro dispositivo.

50 En el ejemplo de la FIG. 1, el dispositivo de origen 12 puede incluir una fuente de vídeo 20, un codificador de vídeo 22, un modulador/demodulador (módem) 23 y un transmisor 24. El dispositivo de destino 16 puede incluir un receptor 26, un módem 27, un decodificador de vídeo 28 y un dispositivo de visualización 30. De acuerdo con la presente divulgación, el codificador de vídeo 22 del dispositivo de origen 12 puede estar configurado para aplicar una o más de las técnicas de la presente divulgación como parte de un proceso de codificación de vídeo. De manera similar, el decodificador de vídeo 28 del dispositivo de destino 16 puede estar configurado para aplicar una o más de las técnicas de la presente divulgación como parte de un proceso de decodificación de vídeo.

60

65 De nuevo, el sistema ilustrado 10 de la FIG. 1, es simplemente ejemplar. Las diversas técnicas de la presente divulgación pueden realizarse mediante cualquier dispositivo de codificación que admita la codificación predictiva basada en bloques, o mediante cualquier dispositivo de decodificación que admita la decodificación predictiva basada en bloques. El dispositivo de origen 12 y el dispositivo de destino 16 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 16. En algunos casos, los dispositivos 12, 16 pueden funcionar de manera

esencialmente simétrica, de modo que cada uno de los dispositivos 12, 16 incluya componentes de codificación y de decodificación de vídeo. Por lo tanto, el sistema 10 puede dar soporte a una transmisión de vídeo unidireccional o bidireccional entre los dispositivos de vídeo 12, 16, por ejemplo, para la transmisión de vídeo, la reproducción de vídeo, la radiodifusión de vídeo o la videotelefonía.

5 La fuente de vídeo 20 del dispositivo de origen 12 incluye un dispositivo de captación de vídeo, tal como una cámara de vídeo, un archivo de vídeo que contiene vídeo captado previamente o vídeo procedente de un proveedor de contenido de vídeo. Como una alternativa adicional, la fuente de vídeo 20 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 20 es una videocámara, el dispositivo fuente 12 y el dispositivo de destino 16 pueden formar los denominados teléfonos con cámara o videoteléfonos. En cada caso, el vídeo captado, precaptado o generado por ordenador puede codificarse mediante el codificador de vídeo 22. La información de vídeo codificada puede modularse entonces mediante un módem 23 según una norma de comunicación, por ejemplo, tal como acceso múltiple por división de código (CDMA) u otra norma de comunicación, y transmitirse al dispositivo de destino 16 a través del transmisor 24 y el canal de comunicación 15. El módem 23 puede incluir varios mezcladores, filtros, amplificadores u otros componentes diseñados para la modulación de señales. El transmisor 24 puede incluir circuitos diseñados para transmitir datos, incluidos amplificadores, filtros y una o más antenas.

10 El receptor 26 del dispositivo de destino 16 recibe información a través del canal de comunicación 15, y el módem 27 demodula la información. Como el transmisor 24, el receptor 26 puede incluir circuitos diseñados para recibir datos, que incluyen amplificadores, filtros y una o más antenas. En algunos casos, el transmisor 24 y/o el receptor 26 pueden incorporarse dentro de un único componente de transceptor que incluye tanto circuitos de recepción como de transmisión. El módem 27 puede incluir varios mezcladores, filtros, amplificadores u otros componentes diseñados para la demodulación de señales. En algunos casos, los módems 23 y 27 pueden incluir componentes para realizar la modulación y la demodulación.

20 De nuevo, el proceso de codificación de vídeo realizado por el codificador de vídeo 22 puede implementar una o más de las técnicas descritas en el presente documento durante la compensación de movimiento. El proceso de decodificación de vídeo realizado por el decodificador de vídeo 28 puede realizar también dichas técnicas durante su fase de compensación de movimiento del proceso de decodificación. El término "codificador" se utiliza en el presente documento para referirse a un dispositivo o aparato informático especializado que realiza la codificación o la decodificación de vídeo. El término "codificador" se refiere en general a cualquier codificador de vídeo, decodificador de vídeo o codificador/decodificador combinado (códec). El término "codificación" se refiere a la codificación o la decodificación. El dispositivo de visualización 30 muestra los datos de vídeo descodificados a un usuario y puede comprender cualquiera entre una variedad de 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 emisores de luz (OLED) u otro tipo de dispositivo de visualización.

30 En el ejemplo de la FIG. 1, el canal de comunicación 15 puede comprender cualquier medio de comunicación inalámbrico o cableado, tal como un espectro de radiofrecuencia (RF) o una o más líneas de transmisión física, o cualquier combinación de medios inalámbricos y cableados. El canal de comunicación 15 puede formar parte de una red basada en paquetes, tal como una red de área local, una red de área extensa o una red global tal como Internet. El canal de comunicación 15 representa generalmente cualquier medio de comunicación adecuado, o un conjunto de diferentes medios de comunicación, para transmitir datos de vídeo desde el dispositivo de origen 12 hasta el dispositivo de destino 16. El canal de comunicación 15 puede incluir encaminadores, conmutadores, estaciones base o cualquier otro equipo que pueda ser útil para facilitar la comunicación desde el dispositivo de origen 12 al dispositivo de destino 16.

40 El codificador de vídeo 22 y el decodificador de vídeo 28 pueden funcionar de acuerdo con una norma de compresión de vídeo, tal como la norma UIT-T H.264, de forma alternativa descrita como MPEG-4, Parte 10, Codificación avanzada de vídeo (AVC). Sin embargo, las técnicas de la presente divulgación no están limitadas a ninguna norma de codificación de vídeo particular. Aunque no se muestra en la FIG. 1, en algunos aspectos, el codificador de vídeo 22 y el decodificador de vídeo 28 pueden estar integrado, cada uno de ellos, con un codificador y decodificador de audio, y pueden incluir unidades adecuadas de multiplexado y demultiplexado, u otro hardware y software, para gestionar la codificación, tanto de audio como de vídeo, en un flujo de datos común o en flujos de datos diferentes. Si procede, las unidades MUX-DEMUX pueden ajustarse al protocolo de multiplexado UIT H.223 o a otros protocolos, tales como el protocolo de datagramas de usuario (UDP).

50 El codificador de vídeo 22 y el decodificador de vídeo 28 pueden implementarse como uno o más microprocesadores, procesadores de señales digitales (DSP), circuitos integrados específicos de la aplicación (ASIC), matrices de puertas programables in situ (FPGA), lógica discreta, software, hardware, firmware o cualquier combinación de estos. El codificador de vídeo 22 y el decodificador de vídeo 28 pueden incluirse cada uno en uno o más codificadores o decodificadores, donde cualquiera de los cuales puede estar integrado como parte de un códec combinado que proporciona capacidades de codificación y decodificación en un respectivo dispositivo móvil, dispositivo de abonado, dispositivo de radiodifusión, servidor o similares.

Una secuencia de vídeo incluye típicamente una serie de tramas de vídeo. Un codificador de vídeo 22 actúa en bloques de vídeo de tramas de vídeo individuales con el fin de codificar los datos de vídeo. Los bloques de vídeo pueden presentar tamaños fijos o variables y pueden diferir en tamaño de acuerdo con una norma de codificación especificada. Cada trama de vídeo incluye una serie de fragmentos. Cada fragmento puede incluir una serie de macrobloques, que pueden disponerse en subbloques. En un ejemplo, la norma UIT-T H.264 admite la intrapredicción en varios tamaños de bloque, tales como 16 por 16, 8 por 8 o 4 por 4 para componentes de luma, y 8x8 para componentes de croma, así como interpredicción en varios tamaños de bloque, tales como 16 por 16, 16 por 8, 8 por 16, 8 por 8, 8 por 4, 4 por 8 y 4 por 4 para componentes de luma y tamaños escalados correspondientes para componentes de croma. Los bloques de vídeo pueden comprender bloques de datos de píxel, o bloques de coeficientes de transformada, por ejemplo, tras un proceso de transformada tal como de transformada de coseno discreta (DCT) o un proceso de transformada conceptualmente similar.

Bloques de vídeo más pequeños pueden proporcionar una mejor resolución y pueden usarse en ubicaciones de una trama de vídeo que incluyen altos niveles de detalle. En general, los macrobloques y los diversos subbloques pueden considerarse como bloques de vídeo. Además, un fragmento puede considerarse una serie de bloques de vídeo, tales como macrobloques y/o subbloques. Cada fragmento puede ser una unidad independientemente decodificable de una trama de vídeo. De forma alternativa, las propias tramas pueden ser unidades decodificables, o pueden definirse otras partes de una trama como unidades decodificables. El término "unidad codificada" se refiere a cualquier unidad de una trama de vídeo que puede decodificarse de manera independiente, tal como una trama completa, un fragmento de una trama, u otra unidad que pueda decodificarse de manera independiente definida de acuerdo con las técnicas de codificación utilizadas.

Para codificar los bloques de vídeo, el codificador de vídeo 22 realiza intra o interpredicción para generar un bloque de predicción. El codificador de vídeo 22 resta los bloques de predicción de los bloques de vídeo originales que se van a codificar para generar bloques residuales. De este modo, los bloques residuales son indicativos de diferencias de píxel a píxel entre los bloques que se codifican y los bloques de predicción. El codificador de vídeo 22 puede aplicar una transformada a los bloques residuales para generar bloques de coeficientes de transformada. Siguiendo las técnicas de codificación de intra o interpredicción y de transformada, el codificador de vídeo 22 realiza la cuantificación. La cuantificación se refiere en general a un proceso en el que los coeficientes se cuantifican para reducir posiblemente la cantidad de datos utilizados para representar los coeficientes. Después de la cuantificación, la codificación por entropía se puede realizar de acuerdo con una metodología de codificación por entropía, tal como la codificación de longitud variable adaptativa al contexto (CAVLC) o la codificación aritmética binaria adaptativa al contexto (CABAC). Más detalles de cada etapa del proceso de codificación realizado por el codificador de vídeo 22 se describirán con más detalle a continuación en la FIG. 2.

En el dispositivo de destino 16, el decodificador de vídeo 28 recibe los datos de vídeo codificados. El decodificador de vídeo 28 realiza la decodificación por entropía de los datos de vídeo de acuerdo con una metodología de codificación por entropía, tal como la CAVLC o la CABAC, para obtener los coeficientes cuantificados. El decodificador de vídeo 28 aplica las funciones de cuantificación inversa (decuantificación) y de transformada inversa para reconstruir el bloque residual en el dominio del píxel. El decodificador 28 de vídeo genera también un bloque de predicción basado en información de control o información de sintaxis (por ejemplo, modalidad de codificación, vectores de movimiento, sintaxis que define coeficientes de filtro y similares) incluidos en los datos de vídeo codificados. El decodificador de vídeo 28 suma el bloque de predicción con el bloque residual reconstruido para generar un bloque de vídeo reconstruido para su visualización. Más detalles de cada etapa del proceso de codificación realizado por el codificador de vídeo 22 se describirán con más detalle a continuación en la FIG. 16.

De acuerdo con las técnicas de esta divulgación, el codificador de vídeo 22 y el decodificador de vídeo 28 pueden utilizar la una o más técnicas de filtrado de interpolación durante la compensación de movimiento. En particular, de acuerdo con un aspecto de esta divulgación, el codificador de vídeo 22 y/o el decodificador de vídeo 28 puede obtener un bloque de píxeles, en el que el bloque de píxeles incluye valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro del bloque de píxeles, calculan valores de subpíxeles para posiciones de subpíxeles asociadas con el bloque de píxeles en base a los valores de píxeles enteros, en el que el cálculo de los valores de subpíxeles comprende aplicar un filtro de interpolación que define una matriz bidimensional de posiciones de soporte de filtro que corresponde a un conjunto de doce o más posiciones de píxeles enteros que rodean las posiciones de subpíxeles de una forma radial, y generar un bloque de predicción basado en al menos algunos de los valores de subpíxeles. Un ejemplo de la matriz bidimensional de posiciones de soporte de filtro que corresponde a un conjunto de doce o más posiciones de píxeles enteros se explica con mayor detalle a continuación.

De acuerdo con otro aspecto de esta divulgación, el codificador de vídeo 22 y/o el decodificador de vídeo 28 pueden utilizar aspectos de simetría para reducir la cantidad de datos que necesitan para comunicarse entre el dispositivo de origen 12 y el dispositivo de destino 16 para la comunicación de coeficientes de filtro utilizados en la interpolación. El codificador de vídeo 22 puede determinar ocho conjuntos de coeficientes de filtro para quince ubicaciones de subpíxeles diferentes, en el que los ocho conjuntos de coeficientes de filtro se generan basándose en la simetría de coeficientes y la simetría de píxeles entre quince ubicaciones de subpíxeles, y emiten los ocho conjuntos de coeficientes de filtro a otro dispositivo como parte de un flujo de bits codificado. De esta manera, los ocho conjuntos,

5 junto con aspectos de simetría de píxeles y simetría de coeficientes, pueden definir todos los coeficientes de filtro para las quince posiciones de píxeles de medio pel y cuarto de pel. Además, puede existir simetría de píxeles entre diferentes de las quince ubicaciones de subpíxeles en una dimensión vertical y en una dimensión horizontal, pero la simetría de píxeles puede no existir en una dimensión diagonal para al menos algunas de las ubicaciones de subpíxeles de quince conjuntos. Esta falta de simetría de píxeles en la dimensión diagonal para al menos algunas de las quince ubicaciones puede mejorar las interpolaciones y la calidad de vídeo en la codificación y decodificación de vídeo.

10 El decodificador de vídeo 28 del dispositivo de destino 16 puede recibir los ocho conjuntos de coeficientes de filtro como parte de un flujo de bits de vídeo codificado, generar quince conjuntos de coeficientes de filtro correspondientes a quince ubicaciones de subpíxeles diferentes basadas en los ocho conjuntos de coeficientes de filtro, generar datos predictivos interpolados para decodificación de vídeo en base a uno de los quince conjuntos de coeficientes de filtro, en el que los datos predictivos interpolados corresponden a una de las quince ubicaciones de subpíxeles diferentes, y decodificar uno o más bloques de vídeo en base a los datos predictivos interpolados.

15 De acuerdo con otro aspecto de esta divulgación, el codificador de vídeo 22 y/o el decodificador de vídeo 28 pueden utilizar el filtrado de tipo interpolación con respecto a posiciones de píxeles enteros con el fin de generar valores de píxeles enteros ajustados. Este tipo de filtrado de interpolación puede mejorar la compresión específicamente durante los cambios de iluminación, la transición o desvanecimiento de escena, puede eliminar el ruido y facilitar la nitidez del marco de la imagen, y puede ayudar a mejorar la codificación del movimiento del objeto fino entre tramas de vídeo sucesivas, particularmente cuando no se impone simetría en los coeficientes de filtro.

20 Las técnicas de filtrado de tipo interpolación del codificador de vídeo 22 y/o el decodificador de vídeo 28 pueden incluir obtener bloques de píxeles, en las que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles, filtrar los valores de píxeles enteros basados en otros valores de píxeles enteros dentro del bloque de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y generar un bloque de predicción basado en los valores de píxeles enteros ajustados.

25 De acuerdo con otro aspecto de esta divulgación, el codificador de vídeo 22 puede generar primeros datos predictivos interpolados para codificar datos de vídeo basados en un primer filtro de interpolación, generar segundos datos predictivos interpolados para la codificación de vídeo de los datos de vídeo basados en un segundo filtro de interpolación, seleccionar entre los primeros datos predictivos interpolados y los segundos datos predictivos interpolados basados en un análisis de distorsión de velocidad, codificar los datos de vídeo basados en la selección y codificar la sintaxis para indicar la selección. El primer filtro de interpolación puede comprender un filtro de interpolación fijo, y el segundo filtro de interpolación puede comprender un filtro de interpolación adaptativo, pero esta divulgación no está necesariamente limitada a estos ejemplos.

30 Además, también se pueden aplicar filtros de interpolación adicionales para generar datos predictivos interpolados adicionales, que también se pueden considerar en el análisis de distorsión de velocidad. En otras palabras, las técnicas de esta divulgación no se limitan a generar sólo primeros y segundos datos predictivos interpolados basados en dos filtros de interpolación, sino que podrían aplicarse para generar cualquier pluralidad de datos predictivos interpolados basados en cualquier número de filtros de interpolación. Es importante destacar que se usan datos predictivos interpolados por análisis de distorsión de velocidad para identificar qué filtro seleccionar.

35 En un ejemplo, un método puede comprender generar una pluralidad de versiones diferentes de datos predictivos para codificar datos de vídeo basados en una pluralidad de diferentes filtros de interpolación, seleccionar entre la pluralidad de versiones diferentes de datos predictivos basados en un análisis de distorsión de velocidad, codificar los datos de vídeo basados en la selección, y codificar la sintaxis para indicar la selección.

40 Esta divulgación también contempla una técnica para codificar coeficientes de filtro. Por ejemplo, el codificador de vídeo 22 puede identificar un conjunto de coeficientes de filtro para la interpolación de datos predictivos en codificación de vídeo, generar valores residuales asociados con el conjunto de coeficientes de filtro basados en la codificación predictiva del conjunto de coeficientes de filtro relativos a coeficientes de filtro asociados a un filtro de interpolación fijo, aplicar la cuantificación a los valores residuales, y emitir los valores residuales cuantificados como parte de un flujo de bits codificado.

45 El decodificador de vídeo 28 puede recibir valores residuales asociados con un conjunto de coeficientes de filtro, generar el conjunto de coeficientes de filtro usando decodificación predictiva basándose en el conjunto de valores residuales y coeficientes de filtro asociados a un filtro de interpolación fijo, y aplicar el conjunto de coeficientes de filtro para interpolar datos predictivos utilizados para la decodificación predictiva de bloques de vídeo.

50 La FIG. 2 es un diagrama de bloques que ilustra un ejemplo de un codificador de vídeo 50 que puede realizar las técnicas de filtrado coherentes con esta divulgación. El codificador de vídeo 50 es un ejemplo de dispositivo o aparato informático de vídeo, denominado "codificador" en el presente documento. El codificador de vídeo 50 puede corresponder al codificador de vídeo 22 del dispositivo 20, o a un codificador de vídeo de un dispositivo diferente. El

codificador de vídeo 50 puede realizar intra e intercodificación bloques dentro de tramas de vídeo, aunque los componentes de intracodificación no se muestran en la FIG. 2 a fin de facilitar la ilustración. La intracodificación se basa en la predicción espacial para reducir o eliminar la redundancia espacial en el vídeo de una trama de vídeo dada. La intercodificación se basa en la predicción temporal para reducir o eliminar la redundancia temporal en el vídeo dentro de tramas adyacentes de una secuencia de vídeo. El modo intra (modo I) puede referirse al modo de compresión con base espacial, y los modos inter, tal como de predicción (modo P) o bidireccional (modo B) pueden referirse a los modos de compresión con base temporal. Las técnicas de esta divulgación se aplican durante la intercodificación, y por lo tanto, las unidades de intracodificación, tales como la unidad de predicción espacial, no se ilustran en la FIG. 2 con fines de simplicidad y facilidad de ilustración.

Como se muestra en la FIG. 2, el codificador de vídeo 50 recibe un bloque de vídeo dentro de una trama de vídeo a codificar. En el ejemplo de la FIG. 2, el codificador de vídeo 50 incluye una unidad de predicción 32, una memoria 34, un sumador 48, una unidad de transformada 38, una unidad de cuantificación 40 y una unidad de codificación por entropía 46. Para la reconstrucción de bloques de vídeo, el codificador de vídeo 50 incluye además una unidad de cuantificación inversa 42, una unidad de transformación inversa 44 y un sumador 51. También puede incluirse un filtro de desbloqueo (no mostrado) para filtrar límites de bloque a fin de eliminar artefactos de pixelado del vídeo reconstruido. Si se desea, el filtro de desbloqueo filtrará típicamente la salida del sumador 51.

La unidad de predicción 32 puede incluir una unidad de estimación de movimiento (ME) 35 y una unidad de compensación de movimiento (MC) 37. El filtro 37 puede incluirse en la unidad de predicción 32, y una o ambas de las unidades de ME 35 y MC 37 pueden recurrir a éste para realizar la interpolación o filtrado de tipo interpolación como parte de la estimación de movimiento y/o la compensación de movimiento, de acuerdo con esta divulgación. El filtro 37 puede representar en realidad una pluralidad de filtros diferentes para facilitar numerosos tipos diferentes de interpolación y filtrado tipo interpolación como se describe en el presente documento. Por lo tanto, la unidad de predicción 32 puede incluir una pluralidad de filtros de interpolación o de filtros tipo interpolación. Durante el proceso de codificación, el codificador de vídeo 50 recibe un bloque de vídeo a codificar (denominado "BLOQUE DE VÍDEO" en la FIG. 2), y la unidad de predicción 32 realiza la codificación entre predicciones para generar un bloque de predicción (denominado "BLOQUE DE PRED." en la FIG. 2). Específicamente, la unidad de ME 35 puede realizar una estimación de movimiento para identificar el bloque de predicción en la memoria 34, y la unidad de MC 37 puede realizar una compensación de movimiento para generar el bloque de predicción.

Se considera típicamente que la estimación de movimiento es el proceso de generar vectores de movimiento, que estiman un movimiento para bloques de vídeo. Un vector de movimiento, por ejemplo, puede indicar el desplazamiento de un bloque de predicción de una trama de predicción o referencia (u otra unidad codificada, por ejemplo, un fragmento) con respecto al bloque que se va a codificar en la trama actual (u otra unidad codificada). La trama de referencia (o parte de la trama) puede estar situada temporalmente antes o después de la trama de vídeo (o parte de la trama de vídeo) a la que pertenece el bloque de vídeo actual. La compensación de movimiento se considera típicamente el proceso de extraer o generar el bloque de predicción de la memoria 34, o posiblemente interpolar o generar de otro modo los datos predictivos filtrados, basándose en el vector de movimiento determinado mediante la estimación de movimiento.

La unidad de ME 35 selecciona el vector de movimiento apropiado para el bloque de vídeo que se va a codificar, comparando el bloque de vídeo con los bloques de vídeo de una o más tramas de referencia (por ejemplo, una trama previa y/o posterior). La unidad de ME 35 puede realizar la estimación de movimiento con precisión de píxel fraccionario, a veces denominada estimación de movimiento de píxel fraccionario, pel fraccionario, o subpíxel. Como tal, las expresiones estimación de movimiento de píxel fraccionario, pel fraccionario y subpíxel pueden usarse indistintamente. En la estimación de movimiento de píxel fraccionario, la unidad de ME 35 puede seleccionar un vector de movimiento que indica un desplazamiento a una ubicación distinta de una ubicación de píxel entero. De esta manera, la estimación de movimiento de píxel fraccionario permite a la unidad de predicción 32 rastrear el movimiento con mayor precisión que las ubicaciones de píxel entero (o píxel completo) y, por tanto, generar un bloque de predicción más preciso. La estimación de movimiento de píxel fraccionario puede tener una precisión de medio píxel, una precisión de un cuarto de píxel, una precisión de un octavo de píxel o cualquier precisión más fina. La unidad de ME 35 puede recurrir a uno o más filtros 39 para cualquier interpolación necesaria durante el proceso de estimación de movimiento.

Para realizar una compensación de movimiento de píxel fraccionario, la unidad MC 37 puede realizar una interpolación (a veces denominada filtrado de interpolación) para generar datos a una resolución de subpíxeles (denominada en el presente documento como valores de subpíxeles o de píxel fraccionario). La unidad MC 37 puede recurrir a uno o más 39 para esta interpolación. La unidad de predicción 32 puede realizar la interpolación (o filtrado de tipo interpolación de píxeles enteros) usando las técnicas descritas en el presente documento.

Una vez seleccionado el vector de movimiento para el bloque de vídeo a codificar por la unidad ME 35, la unidad MC 37 genera el bloque de vídeo de predicción asociado con ese vector de movimiento. La unidad MC 37 puede extraer el bloque de predicción de la memoria 34 en base al vector de movimiento determinado por la unidad ME 35. En el caso de un vector de movimiento con precisión de píxel fraccionario, la unidad MC 37 filtra los datos de la memoria 34 para interpolar dichos datos a una resolución de subpíxeles, por ejemplo, recurriendo a uno o más filtros 39 para

este proceso. En algunos casos, la técnica o modo de filtrado de interpolación que se utilizó para generar los datos de predicción de subpíxeles se puede indicar como uno o más elementos de sintaxis de interpolación para la unidad de codificación de entropía 46 para su inclusión en el flujo de bits codificado. De hecho, algunos aspectos de esta divulgación se refieren al uso de simetría de píxeles y simetría de coeficientes para reducir la cantidad de sintaxis que se necesita transmitir.

Una vez que la unidad de predicción 32 ha generado el bloque de predicción, el codificador de vídeo 50 forma un bloque de vídeo residual (denominado "BLOQUE RESID." en la FIG. 2) restando el bloque de predicción del bloque de vídeo original que está siendo codificado. El sumador 48 representa el componente o los componentes que realizan esta operación de sustracción. La unidad de transformada 38 aplica una transformada, tal como una transformada discreta del coseno (DCT) o una transformada conceptualmente similar, al bloque residual, generando un bloque de vídeo que comprende coeficientes de bloque de transformada residual. La unidad de transformada 38 puede llevar a cabo otras transformadas, tales como las definidas por la norma H.264, que son conceptualmente similares a la DCT. También podrían usarse transformaciones de ondículas, transformaciones de enteros, transformaciones de subbandas u otros tipos de transformaciones. En cualquier caso, la unidad de transformada 38 aplica la transformada al bloque residual, generando un bloque de coeficientes de transformada residuales. La transformada puede convertir la información residual de un dominio de píxel en un dominio de frecuencia.

La unidad de cuantificación 40 cuantifica los coeficientes de transformada residuales para reducir todavía más la velocidad de bits. El proceso de cuantificación puede reducir la profundidad de bits asociada a algunos o todos los coeficientes. Tras la cuantificación, la unidad de codificación por entropía 46 realiza la codificación por entropía de los coeficientes de transformada cuantificados. Por ejemplo, la unidad de codificación por entropía 46 puede realizar la CAVLC, la CABAC u otro procedimiento de codificación por entropía.

La unidad de codificación por entropía 46 también puede codificar uno o más elementos de sintaxis de predicción obtenidos a partir de la unidad de predicción 32 u otro componente del codificador de vídeo 50. El uno o más elementos de sintaxis de predicción pueden incluir un modo de codificación, uno o más vectores de movimiento, una técnica de interpolación que se utilizó para generar los datos de subpíxeles, un conjunto o subconjunto de coeficientes de filtro, u otra información asociada con la generación del bloque de predicción. La unidad de predicción y cuantificación de coeficientes 41 puede codificar y cuantificar de forma predictiva la sintaxis de predicción, tal como coeficientes de filtro, de acuerdo con algunos aspectos de esta divulgación. Tras la codificación por entropía realizada por la unidad de codificación por entropía 46, el vídeo codificado y los elementos sintácticos pueden transmitirse a otro dispositivo o archivarlos para su posterior transmisión o recuperación.

La unidad de cuantificación inversa 42 y la unidad de transformación inversa 44 aplican la cuantificación inversa y la transformación inversa, respectivamente, para reconstruir el bloque residual en el dominio de píxeles, por ejemplo, para su uso posterior como un bloque de referencia. El bloque residual reconstruido (denominado "BLOQUE RESID. RECON." en la FIG. 2) puede representar una versión reconstruida del bloque residual proporcionado a la unidad de transformada 38. El bloque residual reconstruido puede diferir del bloque residual generado por el sumador 48, debido a la pérdida de detalle causada por las operaciones de cuantificación y cuantificación inversa. El sumador 51 suma el bloque residual reconstruido al bloque de predicción con compensación de movimiento, generado por la unidad de predicción 32, para generar un bloque de vídeo reconstruido para su almacenamiento en la memoria 34. La unidad de predicción 32 puede utilizar el bloque de vídeo reconstruido como bloque de referencia que se puede utilizar para codificar posteriormente un bloque en una trama de vídeo subsiguiente o una unidad codificada posterior.

Como se ha descrito anteriormente, la unidad de predicción 32 puede realizar la estimación de movimiento con una precisión de píxel fraccionario (o subpíxel). Cuando la unidad de predicción 32 utiliza una estimación de movimiento de píxeles fraccionada, la unidad de predicción 32 puede generar datos a una resolución de subpíxeles (por ejemplo, valores de píxeles de subpíxel o fraccionarios) utilizando operaciones de interpolación descritas en esta divulgación. En otras palabras, las operaciones de interpolación se utilizan para calcular valores en posiciones entre las posiciones de píxeles enteros. Las posiciones de subpíxel situadas a mitad de la distancia entre las posiciones de píxeles enteros pueden denominarse posiciones de medio píxel (medio píxel), las posiciones de subpíxel situadas a mitad de distancia entre una posición de píxeles enteros y una posición de medio píxel pueden referirse a las posiciones de cuarto de píxel (cuarto de píxel), las posiciones de subpíxel situadas a mitad de la distancia entre una posición de píxeles enteros (o posición de medio píxel) y una posición de cuarto de píxel se denominan posiciones de un octavo de píxel (un octavo de pel), y similares.

La FIG. 3 es un diagrama conceptual que ilustra posiciones de píxeles enteros (o píxeles completos) asociadas con datos de predicción, y posiciones de subpíxeles (o píxel fraccionario) asociadas con datos de predicción interpolados. En la ilustración conceptual de la FIG. 3, los diferentes cuadros representan ubicaciones o posiciones de píxeles y subpíxeles dentro de una trama o un bloque de una trama. Las letras mayúsculas (en los cuadros con líneas continuas) representan ubicaciones de píxeles enteros, mientras que las letras pequeñas (en los cuadros con líneas discontinuas) representan las ubicaciones de subpíxeles. En particular, las ubicaciones de píxeles A1-A6, B1-B6, C1-C6, D1-D6, E1-E6 y F1-F6 representan una matriz de 6 por 6 de posiciones de píxeles enteros dentro de un marco, un fragmento u otra unidad codificada. Las ubicaciones de subpíxeles "a" a "o" representan quince

ubicaciones de subpíxeles asociadas con el píxel entero C3, por ejemplo, entre ubicaciones de píxeles enteros C3, C4, D3 y D4. Pueden existir ubicaciones de subpíxeles similares para cada posición de píxel entero. Las ubicaciones de los subpíxeles "a" a "o" representan cada posición de píxel de medio pel y cuarto de pel asociada con el píxel entero C3.

5 Las ubicaciones de píxeles enteros pueden estar asociadas con un elemento sensor físico, tal como un fotodiodo cuando los datos de vídeo se generaron originalmente. El fotodiodo puede medir una intensidad de una fuente de luz en la ubicación del sensor y asociar un valor de intensidad de píxel con la posición de píxel entero. De nuevo, cada ubicación de píxeles enteros puede tener un conjunto asociado de quince ubicaciones de subpíxeles (o posiblemente más). El número de ubicaciones de subpíxeles asociado con ubicaciones de píxeles enteros puede depender de la precisión deseada. En el ejemplo ilustrado en la FIG. 3, la precisión deseada es la precisión de un cuarto de píxel, en cuyo caso, cada una de las ubicaciones de píxeles enteros corresponde con quince posiciones diferentes de subpíxeles. Pueden asociarse más o menos posiciones de subpíxeles a cada ubicación de píxel entero en base a la precisión deseada. Para una precisión de medio píxel, por ejemplo, cada ubicación de píxel entero puede corresponder con tres posiciones de subpíxel. Como otro ejemplo, cada una de las ubicaciones de píxeles enteros puede corresponder con sesenta y tres posiciones de subpíxel para una precisión de un octavo de píxel. Cada ubicación de píxel puede definir uno o más valores de píxel, por ejemplo, uno o más valores de luminancia y crominancia.

20 Y puede representar luminancia, y Cb y Cr pueden representar dos valores diferentes de crominancia de un espacio de color YCbCr tridimensional. Cada ubicación de píxel puede definir realmente tres valores de píxeles para un espacio de color tridimensional. Sin embargo, las técnicas de la presente divulgación pueden referirse a la predicción con respecto a una dimensión a fin de simplificar. En la medida en que se describen técnicas con respecto a los valores de píxel en una dimensión, se pueden ampliar técnicas similares a las otras dimensiones.

25 En el ejemplo de la FIG. 3, las ubicaciones de subpíxeles asociadas con el píxel entero "C3" se ilustran para una precisión de un cuarto de píxel. Las quince posiciones de subpíxel asociadas con el píxel C3 están etiquetadas como "a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l", "m", "n", y "o". La mayoría de las otras ubicaciones fraccionarias asociadas con otras ubicaciones de píxeles enteros no se muestran por simplicidad (distintas de las usadas para generar una o más de las 15 ubicaciones fraccionarias diferentes asociadas con la ubicación de píxeles C3, como se describe con más detalle a continuación). Las ubicaciones de subpíxeles "b", "h" y "j" se pueden denominar ubicaciones de medio píxel y las ubicaciones de subpíxeles "a", "c", "d", "e", "f", "g", "i", "k", "l", "m", y "o" se pueden denominar ubicaciones de un cuarto de píxel.

35 La unidad de predicción 32 del codificador de vídeo 40 puede determinar valores de píxel para las ubicaciones de subpíxeles "a" a "o" utilizando el filtrado de interpolación por la unidad MC 37. De acuerdo con la norma ITU-T H.264, por ejemplo, la unidad de predicción 32 puede determinar valores de píxeles para ubicaciones de medio píxel usando un filtro de interpolación de 6 taps, tal como un filtro Wiener. En el caso de la norma H.264, los coeficientes de filtro para el filtro de interpolación de 6 taps son típicamente [1, -5, 20, 20, -5, 1], aunque pueden utilizarse otros coeficientes. La unidad de predicción 32 puede aplicar el filtro de interpolación primero en la dirección horizontal y luego en la dirección vertical, o viceversa. Para posiciones de medio píxel "b" y "h", cada tap puede corresponder a una posición de píxel entero en la dirección horizontal y vertical, respectivamente. En particular, para la posición "b" de medio píxel, los taps del filtro de 6 taps corresponden a C1, C2, C3, C4, C5 y C6. Del mismo modo, para la posición de medio píxel "h", los taps del filtro de 6 taps corresponden a A3, B3, C3, D3, E3 y F3. Por ejemplo, los valores de píxel para las posiciones de subpíxel "b" y "h" se pueden calcular usando las ecuaciones (1) y (2):

$$b = ((C1 - 5*C2 + 20*C3 + 20*C4 - 5*C5 + C6) + 16)/32 \quad (1)$$

$$h = ((A3 - 5*B3 + 20*C3 + 20*D3 - 5*E3 + F3) + 16)/32 \quad (2)$$

50 Para la posición de medio píxel "j", los taps del filtro de 6 taps corresponden a posiciones interpoladas horizontalmente entre las posiciones C1-C6 y D1-D6, o verticalmente entre las posiciones A3-F3 y A4-F4. La posición de medio píxel "j" se puede calcular con un filtro de 6 taps que utiliza valores de píxeles previamente interpolados de las posiciones de medio píxel, por ejemplo, de acuerdo con una de las ecuaciones (3) o (4):

$$j = ((aa - 5*bb + 20*b + 20*hh - 5*ii + jj) + 16)/32 \quad (3)$$

$$j = ((cc - 5*dd + 20*h + 20*ee - 5*ff + gg) + 16)/32 \quad (4)$$

60 donde (como se ilustra en la FIG. 3) aa corresponde a una interpolación entre A3 y A4, bb corresponde a una interpolación entre B3 y B4, b corresponde a una interpolación entre C3 y C4, hh corresponde a una interpolación entre D3 y D4, ii corresponde a una interpolación entre E3 y E4 y jj corresponde a una interpolación entre F3 y F4. En la ecuación 4, cc corresponde a una interpolación entre C1 y D1, dd corresponde a una interpolación entre C2 y

D2, h corresponde a una interpolación entre C3 y D3, ee corresponde a una interpolación entre C4 y D4, ff corresponde a una interpolación entre C5 y D5 y gg corresponde a una interpolación entre C6 y D6.

De acuerdo con el estándar H.264, la unidad de predicción 32 puede determinar valores de píxeles en ubicaciones de cuarto de píxel "a", "c", "d", "e", "f", "g", "i", "k", "l", "m", "n" y "o" utilizando un filtro de interpolación bilineal y los valores de píxeles de las ubicaciones circundantes de medio píxel y entero. Por ejemplo, la unidad de predicción 32 puede determinar un valor de píxel asociado con la posición de subpíxel "a" usando valores de píxel de C3 y "b", determinar un valor de píxel asociado con la posición de subpíxel "c" usando valores de píxel de "b" y C4, y similares.

El filtro real que es aplicado por la unidad MC 37 para generar datos interpolados en las ubicaciones de subpíxeles puede estar sujeto a una amplia diversidad de implementaciones. Como un ejemplo, la unidad de predicción 32 puede usar el filtrado adaptativo de interpolación (AIF), como se describe a continuación, para definir los valores interpolados. El comité ITU-T SG16/Q.6/VCEG (Grupo de expertos en codificación de vídeo) ha estado explorando tecnologías de codificación que ofrecen mayor eficiencia de codificación que H.264 y, en particular, AIF. AIF ofrece una gran ganancia de codificación sobre el filtrado de interpolación utilizado en el estándar H.264, especialmente en secuencias de vídeo de alta resolución (por ejemplo, 720i/p o 1080i/p). En AIF, el filtro de interpolación para cada posición de subpíxel se calcula analíticamente para cada trama de vídeo minimizando la energía de error de predicción. Esto ayuda a abordar el solapamiento, la cuantificación y los errores de estimación de movimiento, el ruido de la cámara u otro artefacto contenido en las tramas de vídeo originales y de referencia. Los coeficientes de filtro adaptativo derivados analíticamente para cada trama entonces se predicen, se cuantifican, se codifican y se envían en el flujo de bits de vídeo. Algunas de las técnicas de esta divulgación podrían funcionar dentro de un esquema AIF, así como muchos otros esquemas de interpolación.

Existen muchos tipos diferentes de esquemas AIF coherentes con aspectos de esta divulgación. Por ejemplo, un primer esquema es un AIF bidimensional no separable (NS-AIF), un segundo es un AIF separable (S-AIF), y un tercero es un AIF con filtros direccionales (D-AIF). Aunque cada uno de estos esquemas AIF utiliza diferentes técnicas de interpolación y soporte, los tres esquemas AIF pueden usar procesos analíticos similares para derivar los coeficientes del filtro, lo cual se explica a continuación usando AIF no separable como ejemplo.

Supongamos que un filtro bidimensional no separable de 6 x 6 tiene los coeficientes $h_{i,j}^{SP}$ donde $i,j = 0...5$ y SP representa una de las 15 posiciones de subpíxel ("a" a "o") mostradas en la FIG. 3. Cabe apreciar que 6 de las 15 posiciones de Wsubpíxel, es decir, "a", "b", "c", "d", "h" y "l", son posiciones de subpíxeles unidimensionales (1D), y la unidad de predicción 32 puede usar un filtro de interpolación de 6 taps para interpolar dichos datos. Las posiciones de subpíxel "a", "b", "c", "d", "h" y "l", son 1D en el sentido de que están situadas en una línea horizontal o vertical entre dos posiciones de píxeles enteros. Además, se asume que los píxeles de predicción en las posiciones de píxeles enteros (A1 a F6 en la FIG. 3) en la trama de referencia toman los valores de píxeles de $P_{i,j}$, donde $i,j = 0...5$. Es decir, A1 toma el valor de $P_{0,0}$, ..., A6 toma el valor de $P_{5,0}$, ..., F1 toma el valor de $P_{5,0}$, ..., y F6 toma el valor de $P_{5,5}$. Después, el valor interpolado p^{SP} en la posición de subpíxel SP, $SP \in \{a,...,o\}$, se puede calcular mediante la unidad de predicción 32 usando la siguiente ecuación

$$p^{SP} = \sum_{i=0}^5 \sum_{j=0}^5 P_{i,j} h_{i,j}^{SP} \quad (5)$$

Se espera que $S_{x,y}$ sea el valor de píxel en la trama de vídeo actual en la posición (x, y).

$$\tilde{x} = x + \lfloor mvx \rfloor - FO, \quad \tilde{y} = y + \lfloor mvy \rfloor - FO,$$

donde (mvx,mvy) es el vector de movimiento, (Lmvx|,Lmvy|) es la componente entera del vector de movimiento, y FO es el desfase de filtro. El valor (\tilde{x},\tilde{y}) es la posición de píxel correspondiente en las tramas de referencia. Por ejemplo, en el caso del filtro de 6 taps, $FO = 6/2 - 1 = 2$. Para cada posición de subpíxel SP, la energía de error de predicción (e^{SP})² entre el valor de píxel real en la trama actual y el valor interpolado puede acumularse por la unidad de predicción 32 para todos los píxeles que tienen precisión de vector de movimiento correspondiente a la posición de subpíxel SP. La energía de error de predicción (e^{SP})² puede calcularse mediante la unidad de predicción 32 usando la siguiente ecuación:

$$(e^{SP})^2 = \sum_x \sum_y (S_{x,y} - p_{xy}^{SP})^2 = \sum_x \sum_y \left(S_{x,y} - \sum_{i=0}^5 \sum_{j=0}^5 h_{i,j}^{SP} P_{\tilde{x}+i,\tilde{y}+j} \right)^2 \quad (6)$$

Para cada una de las posiciones de subpíxel a a o, la unidad MC 37 puede establecer un conjunto individual de ecuaciones calculando la derivada de (e^{SP})² con respecto a los coeficientes de filtro $h_{i,j}^{SP}$. El número de ecuaciones, en este caso, es igual al número de coeficientes de filtro utilizados para la posición de subpíxel actual SP. Para cada

posición de subpíxel bidimensional (2D) "e", "f", "g", "i", "j", "k", "m", "n", y "o", la unidad 32 puede usar un filtro de interpolación bidimensional de 6 por 6 taps. Las posiciones de subpíxel "e", "f", "g", "i", "j", "k", "m", "n", y "o", son bidimensionales en el sentido de que no se sitúan en una línea vertical o línea horizontal entre dos posiciones de píxeles enteros. En este caso, un sistema de treinta y seis ecuaciones con treinta y seis incógnitas puede ser resuelto por la unidad MC 37. Las posiciones de subpíxel monodimensionales restantes "a", "b", "c", "d", "h", y "l" sólo requieren un filtro de interpolación monodimensional (por ejemplo, filtro de 6 taps monodimensional). En el caso de un filtro monodimensional de 6 taps, un sistema de seis ecuaciones puede ser resuelto por la unidad MC 37.

$$0 = \frac{(\partial e^{SP})^2}{\partial h_{k,l}^{SP}}$$

$$= \frac{\partial}{\partial h_{k,l}^{SP}} \left(\sum_x \sum_y \left(S_{x,y} - \sum_i \sum_j h_{i,j}^{SP} P_{\bar{x}+i, \bar{y}+j} \right)^2 \right) \quad (7)$$

$$= \sum_x \sum_y \left(S_{x,y} - \sum_i \sum_j h_{i,j}^{SP} P_{\bar{x}+i, \bar{y}+j} \right) P_{\bar{x}+k, \bar{y}+l}$$

$\forall k, l \in \{0;5\}$

El filtro o filtros 39 puede representar un filtro o un conjunto de muchos filtros diferentes que pueden ser utilizados por la unidad MC 37 para generar los datos predictivos.

De este modo, un proceso ejemplar para derivar y aplicar los filtros AIF puede tener las siguientes etapas, que pueden realizarse por la unidad de predicción 32:

1. Estimar los vectores de movimiento (mvx, mvy) para cada bloque de vídeo a codificar. Durante la estimación de movimiento, puede aplicarse un filtro de interpolación fijo (por ejemplo, el proceso de interpolación de H.264/AVC).
2. Usar estos vectores de movimiento, acumular energía de error de predicción para cada posición de subpíxel SP sobre la trama de vídeo actual. Después, calcular los coeficientes de filtro adaptativo $h_{i,j}^{SP}$ para cada posición de subpíxel SP independientemente minimizando la energía de error de predicción como en las dos ecuaciones de energía de predicción anteriores.
3. Estimar nuevos vectores de movimiento. Durante este proceso de estimación de movimiento, pueden aplicarse los filtros de interpolación adaptativa calculados en la etapa 2. Utilizando los filtros de interpolación adaptativa, se reducen los errores de estimación de movimiento, causados por el solapamiento, el ruido de la cámara, etc., y se consigue una mejor predicción de movimiento.

Diferentes esquemas AIF pueden utilizar el mismo proceso analítico que el dado anteriormente. Las diferencias entre los diferentes esquemas radican principalmente en el número de coeficientes de filtro únicos usados, ya sean los filtros de interpolación son separables o no separables, y se utilice el soporte de filtro (es decir, posiciones de píxeles enteros usadas para interpolar al menos una porción de subpíxeles). En cada uno de estos esquemas, se pueden imponer ciertas restricciones de simetría en los filtros AIF para reducir el número de coeficientes de filtro que necesitan codificarse y enviarse en el flujo de bits de vídeo.

Para NS-AIF, por ejemplo, la unidad MC 37 de la unidad de predicción 32 puede interpolar posiciones de subpíxel monodimensionales "a", "b", "c", "d", "h", y "l" usando un interpolación filtro monodimensional de 6 taps (también conocido como un filtro de 6 posiciones ya que cada tap corresponde a una posición de píxel entero), que requiere seis coeficientes. Los seis coeficientes de los filtros de 6 posiciones usados para interpolar el subpíxel monodimensional corresponden cada uno a una de las posiciones de píxeles enteros ilustradas en la FIG. 3. Por ejemplo, para las posiciones de subpíxel "a", "b" y "c", las seis posiciones de píxeles enteros correspondientes a los coeficientes son C1, C2, C3, C4, C5 y C6, y para las posiciones de subpíxel "d", "h" y "l", las seis posiciones de píxeles enteros correspondientes a los coeficientes son A3, B3, C3, D3, E3 y F3. Estas posiciones de píxeles enteros representan el "soporte de filtro" del filtro de interpolación.

La unidad de predicción 32 puede interpolar las posiciones de subpíxel bidimensionales "e", "f", "g", "i", "j", "k", "m", "n", y "o" usando un filtro de interpolación bidimensional de 6 por 6, que requiere treinta y seis coeficientes de filtro. Los treinta y seis coeficientes del filtro de interpolación bidimensional de 6 x 6 utilizados para interpolar el subpíxel bidimensional corresponden cada uno a las posiciones de los píxeles enteros A1-A6, B1-B6, C1-C6, D1-D6, E1-E6 y F1-F6. Estas posiciones de píxeles enteros representan el "soporte de filtro" del filtro de interpolación. Si no se impone ninguna restricción adicional, por ejemplo, sin coeficiente o simetría de píxeles, el codificador de vídeo puede codificar y transmitir nueve conjuntos de treinta y seis coeficientes para las posiciones de subpíxel

bidimensionales y seis conjuntos de seis coeficientes para las posiciones de subpíxel monodimensionales, para un total de 360 coeficientes. La codificación y el envío de ese número de coeficientes en el flujo de bits de vídeo puede dar lugar a costosas sobrecargas de bits. La alta sobrecarga de bits puede, a su vez, aumentar la velocidad de bits para un nivel dado de distorsión, lo cual es indeseable.

5 Para reducir la sobrecarga de bits asociada con el envío de los coeficientes de filtro, pueden imponerse ciertas restricciones de simetría a los filtros de interpolación para reducir el número de coeficientes de filtro únicos que necesitan ser enviados al decodificador 28. Pueden imponerse dos tipos de simetría, es decir, simetría de píxeles y simetría de coeficientes, en solitario o en combinación. La simetría de píxeles impone el mismo conjunto de coeficientes de filtro (y las versiones reflejadas, volteadas y/o giradas del conjunto) para diferentes posiciones de subpíxeles. La simetría de píxeles también puede denominarse simetría de subpíxeles en la medida en que dicha simetría de píxeles se aplique con respecto a los coeficientes de filtro asociados con dos o más ubicaciones de subpíxeles. La simetría de coeficientes, por otra parte, hace que un filtro de interpolación dado sea simétrico en una cierta dirección (por ejemplo, dirección horizontal, dirección vertical o ambas) para las diversas posiciones de soporte de filtro con respecto a otras posiciones de soporte de filtro para un determinado valor de subpíxel a interpolar.

La simetría de píxeles se puede utilizar en NS-AIF. Con referencia de nuevo a la FIG. 3, se espera que $h_0^a, h_1^a, \dots, h_5^a$ sea el conjunto de coeficientes de filtro para la posición de subpíxel "a", entonces el conjunto de coeficientes de filtro para la posición de subpíxel "c" es $h_5^a, h_4^a, \dots, h_0^a$, es decir, el mismo coeficiente en orden inverso o volteado horizontalmente. Es decir, el valor de píxel de predicción pa en la posición de subpíxel "a" y el valor de píxel de predicción pc en la posición de subpíxel "c" se pueden calcular usando (8) y (9), respectivamente.

$$p^a = h_0^a \cdot C_1 + h_1^a \cdot C_2 + h_2^a \cdot C_3 + h_3^a \cdot C_4 + h_4^a \cdot C_5 + h_5^a \cdot C_6 \quad (8)$$

$$p^c = h_5^a \cdot C_1 + h_4^a \cdot C_2 + h_3^a \cdot C_3 + h_2^a \cdot C_4 + h_1^a \cdot C_5 + h_0^a \cdot C_6 \quad (9)$$

25 Las posiciones de subpíxel "d" y "l" pueden interpolarse utilizando el mismo conjunto de coeficientes de filtro de interpolación que las posiciones de subpíxeles "a" y "c", respectivamente. Como tal, las posiciones de subpíxeles "a" y "c" pueden tener simetría de píxeles con respecto a las posiciones de subpíxeles "d" y "l".

30 Un aspecto de esta divulgación es eliminar realmente la simetría de píxeles entre las posiciones de subpíxeles "a" y "c" en relación con las posiciones de subpíxeles "d" y "l". Además, se puede evitar la simetría de píxeles para la posición de subpíxeles "f" con respecto a la posición de subpíxel "i". Además, se puede evitar la simetría de píxeles para la posición de subpíxel "k" con relación a la posición de subpíxel "n". En tales casos, la correlación diagonal puede ser baja, lo que hace ineficiente o ineficaz para imponer simetría diagonal en estos casos.

35 Como otro ejemplo, se espera que $h_{0,0}^e, h_{0,1}^e, \dots, h_{0,5}^e, \dots, h_{5,0}^e, h_{5,1}^e, \dots, h_{5,5}^e$ sea el conjunto de coeficientes de filtro bidimensional de 6 por 6 para la posición de subpíxel "e". Entonces el conjunto de coeficientes de filtro para la posición "g" es $h_{0,5}^e, h_{0,4}^e, \dots, h_{0,0}^e, \dots, h_{5,5}^e, h_{5,4}^e, \dots, h_{5,0}^e$ (versión horizontalmente volteada). De manera similar, el conjunto de coeficientes de filtro para la posición de subpíxel "m" es $h_{5,0}^e, h_{5,1}^e, \dots, h_{5,5}^e, \dots, h_{0,0}^e, h_{0,1}^e, \dots, h_{0,5}^e$ (versión verticalmente invertida), y el conjunto de coeficientes de filtro para la posición de subpíxel "o" es $h_{5,5}^e, h_{5,4}^e, \dots, h_{5,0}^e, \dots, h_{0,5}^e, h_{0,4}^e, \dots, h_{0,0}^e$ (primero volteado horizontalmente y después volteado verticalmente). Como tal, las posiciones de subpíxeles "e", "g", "m" y "o" tienen simetría de píxeles. La simetría entre las posiciones de subpíxeles "b" y "h" y entre las posiciones "f", "i", "k" y "n" se imponen de una manera similar a la de los ejemplos dados anteriormente. Bajo tal restricción de simetría de píxeles, quedan sólo cinco grupos de conjuntos de filtros únicos, un primer conjunto de coeficientes de filtro para el grupo de subpíxeles "a", "c", "d" y "l", un segundo conjunto de coeficientes de filtro para un grupo de subpíxeles "b" y "h", un tercer conjunto de coeficientes de filtro para el grupo de subpíxeles "e", "g", "m" y "o", un cuarto conjunto de coeficientes de filtro para el grupo "f", "i", "k" y "n", y un quinto conjunto de coeficientes de filtro para el grupo de subpíxeles "j".

50 Además, NS-AIF puede imponer restricción de simetría de coeficientes en algunos de estos conjuntos de filtros, en solitario o en combinación con la simetría de subpíxeles descrita anteriormente. En algunos casos, la simetría de coeficientes diagonales puede ser evitada a propósito, por ejemplo, eliminando cualquier restricción de simetría de coeficientes diagonales. En un caso, no se impone ninguna simetría de coeficientes al filtro para el primer grupo de posiciones de subpíxeles, que incluye ubicaciones de subpíxeles "a", "c", "d" y "l". Sin embargo, el segundo grupo de posiciones de subpíxeles, que incluye el coeficiente "b" y "h", puede tener una simetría de coeficientes de acuerdo con la ecuación (10).

$$h_0^b = h_5^b, h_1^b = h_4^b, h_2^b = h_3^b \quad (10)$$

Del mismo modo, los coeficientes para los filtros de interpolación para el tercer, cuarto y quinto grupos de posiciones de subpíxeles, que incluyen posiciones de subpíxel "e", "f" y "j", respectivamente, pueden tener una simetría de coeficientes tal como se da en las ecuaciones (11), (12) y (13), respectivamente.

5

$$h_{i,j}^e = h_{j,i}^e, \text{ para } i, j = 0 \dots 5, i \neq j \quad (11)$$

$$h_{i,j}^f = h_{i,5-j}^f, \text{ para } i, j = 0 \dots 5 \quad (12)$$

10

$$\begin{aligned} h_{i,j}^j = h_{i,5-j}^j = h_{5-i,j}^j = h_{5-i,5-j}^j, \quad i, j = 0 \dots 2 \\ h_{i,j}^j = h_{j,i}^j, \quad \text{para } i \neq j \end{aligned} \quad (13)$$

Bajo estas restricciones de simetría de píxeles y simetría de coeficientes descritas anteriormente, el número de coeficientes de filtro únicos puede reducirse de 360 (sin restricción de simetría) a 6 (a) + 3 (b) + 21 (e) + 18 (f) + 6 (j) = 54 coeficientes, es decir, 6 coeficientes para el grupo que incluye la posición de subpíxel "a", 3 coeficientes para el grupo que incluye la posición de subpíxel "b", 21 coeficientes para el grupo que incluye la posición de subpíxel "e", 18 coeficientes para el grupo que incluye la posición de subpíxel "f", y 6 coeficientes para el grupo que incluye la posición de subpíxel "j". Por lo tanto, en NS-AIF, el codificador de vídeo 22 puede codificar y transmitir cincuenta y cuatro coeficientes en lugar de 360, como en el caso de no tener restricciones de simetría. Como se ha descrito anteriormente, los cincuenta y cuatro coeficientes pueden ser resueltos analíticamente basándose en la ecuación (7) anterior. El codificador de vídeo 50 puede entonces predecir, cuantificar, codificar (por ejemplo, usando el código Exp-Golomb firmado), y enviar los coeficientes en el flujo de bits. Los detalles adicionales sobre la codificación predictiva de coeficientes de filtro se analizan en mayor detalle a continuación. La unidad de predicción y cuantificación de coeficientes 41 puede utilizarse para la codificación predictiva y la cuantificación de coeficientes de filtro consistentes con esta descripción.

25

En otro esquema AIF, es decir, S-AIF, la unidad de predicción 32 puede usar filtros de interpolación separables en la dirección horizontal y en la dirección vertical, en lugar de filtros de interpolación no separables como se usan en NS-AIF. Para las posiciones de subpíxeles monodimensionales, la unidad de predicción 32 (por ejemplo, la unidad MC 37 de la unidad de predicción 32) aplica sólo filtros direccionales horizontales o sólo filtros direccionales verticales dependiendo de la ubicación de subpíxeles. En un ejemplo, los filtros direccionales horizontales y verticales comprenden filtros de 6 posiciones (o 6 taps). La unidad de predicción 32 aplica filtros direccionales horizontales para posiciones de subpíxel "a", "b" y "c" con posiciones de píxeles enteros C1, C2, C3, C4, C5 y C6 (véase la FIG. 3) como soporte de filtro y aplica filtros direccionales verticales para las posiciones de subpíxeles "d", "h" y "l" con las posiciones de píxeles enteros A3, B3, C3, D3, E3 y F3 (véase la FIG. 3) como soporte del filtro. Para las posiciones de subpíxeles restantes, es decir, las posiciones de subpíxeles bidimensionales, la unidad de predicción 32 aplica primero filtrado horizontal, seguido de filtrado vertical o filtrado vertical, seguido de filtrado horizontal. De nuevo, los coeficientes de filtro utilizados para los filtros separables horizontales y verticales se pueden calcular de acuerdo con la ecuación (7) anterior. Sin simetría, S-AIF puede reducir el número de coeficientes que deben codificarse y enviarse a 144 coeficientes a partir de 360 coeficientes, como es el caso de NS-AIF sin simetría.

40

S-AIF puede además imponer una o más restricciones de simetría en al menos algunos de los filtros de interpolación. A diferencia de la simetría de subpíxeles de NS-AIF, con S-AIF, los conjuntos de filtros utilizados para las posiciones de subpíxeles "a" y "c" no son simétricos, es decir, versiones horizontalmente invertidas entre sí. En su lugar, los coeficientes de los conjuntos de filtros utilizados para las posiciones de subpíxeles "a" y "c" se derivan

45

independientemente entre sí como $h_0^a, h_1^a, \dots, h_5^a$ y $h_0^c, h_1^c, \dots, h_5^c$, respectivamente. Para cada posición de subpíxel que tiene movimiento vertical, la simetría de subpíxeles es similar a NS-AIF. Es decir, los coeficientes para las posiciones de subpíxeles "d" y "l", los coeficientes para las posiciones de subpíxeles "e" y "m", los coeficientes para las posiciones de subpíxeles "f" y "n", y los coeficientes para las posiciones de subpíxeles "g" y "o" son versiones volteadas verticalmente entre sí, respectivamente. Por ejemplo, si los coeficientes del filtro vertical para la posición

50

de subpíxel "d" son $h_0^d, h_1^d, \dots, h_5^d$, entonces los coeficientes del filtro vertical para la posición de subpíxel "l" son $h_5^d, h_4^d, \dots, h_0^d$. Por lo tanto, la simetría de subpíxeles utilizada en S-AIF sólo tiene simetría vertical de subpíxeles en una dimensión. Sin embargo, S-AIF no tiene simetría horizontal de subpíxeles. La simetría de subpíxeles en S-AIF puede reducir el número de coeficientes que necesitan ser codificados y enviados de 144 coeficientes a 102 coeficientes.

55

S-AIF, como NS-AIF, también puede utilizar la simetría de coeficientes para reducir aún más el número de coeficientes que necesitan ser codificados y enviados por el codificador de vídeo 50. La simetría de coeficientes utilizada en S-AIF puede ser la misma que en NS-AIF, excepto que en S-AIF sólo hay filtros monodimensionales (horizontales o verticales) de 6 taps. En otras palabras, no hay filtros bidimensionales de 6 por 6. En S-AIF, para

cada posición de subpíxel SP , $SP \in \{b, h, i, j, k\}$, los coeficientes de filtro pueden tener simetría como se indica en la ecuación (14).

$$h_0^{SP} = h_5^{SP}, h_1^{SP} = h_4^{SP}, h_2^{SP} = h_3^{SP} \quad (14)$$

5 Por lo tanto, tanto con simetría de subpíxeles como con simetría de coeficientes, los filtros S-AIF pueden tener once conjuntos de filtros exclusivos monodimensionales de 6 taps, cinco de los cuales son filtros simétricos que tienen cada uno tres coeficientes únicos. Los seis filtros restantes pueden tener cada uno seis coeficientes únicos. En este caso, el codificador de vídeo 22 puede cuantificar, predecir, codificar y enviar un total de cincuenta y un coeficientes de filtro únicos en el flujo de bits de vídeo.

10 En otro esquema AIF, la unidad de predicción D-AIF 32 utiliza filtros direccionales para las posiciones de subpíxeles bidimensionales "e", "f", "g", "i", "j", "k", "m", "n", y "o". Los filtros direccionales pueden comprender filtros de seis taps, y los coeficientes pueden tener simetría de manera que sólo se necesita transmitir un subconjunto de los coeficientes. Como se describe en el presente documento, el soporte de filtro se refiere a las posiciones de píxeles de la trama de referencia que se utiliza para interpolar las posiciones de subpíxeles. La unidad MC 37 puede calcular las posiciones de los subpíxeles "e" y "o" con un filtro de interpolación diagonal de 6 posiciones (o 6 taps) que tiene posiciones de píxeles enteros A1, B2, C3, D4, E5 y F6 como soporte de filtro. La unidad de predicción 32 puede calcular las posiciones de subpíxeles "g" y "m" con un filtro de interpolación diagonal de 6 posiciones que tiene posiciones de píxeles enteros A6, B5, C4, D3, E2 y F1 como soporte de filtro. La unidad de predicción 32 puede calcular las posiciones de subpíxeles "f", "i", "j", "k" y "n" con un filtro de interpolación diagonal de 12 posiciones que tiene posiciones de píxeles enteros A1, B2, C3, D4, E5, F6, A6, B5, C4, D3, E2 y F1 como soporte de filtro.

15 D-AIF puede usar el mismo subpíxel y simetría de coeficientes como se ha descrito anteriormente con respecto a NS-AIF. En cuando a la simetría de subpíxeles, hay cinco conjuntos únicos de coeficientes de filtro para interpolar las quince posiciones de subpíxeles con posiciones de subpíxel "a", "c", "d" y "l" que comparten los mismos coeficientes de filtro (por ejemplo, versiones espiradas, volteadas y/o giradas), posiciones de subpíxeles "e", "g", "m" y "o" que comparten los mismos coeficientes de filtro (por ejemplo, versiones reflejadas, volteadas y/o giradas), posiciones de subpíxeles "b" y "h" que comparten los mismos coeficientes de filtro (siendo "b" un filtro horizontal y siendo "h" un filtro vertical), posiciones de subpíxeles "f", "i", "k" y "n" que comparten los mismos coeficientes de filtro (por ejemplo, versiones reflejadas, volteadas y/o giradas), y la posición de subpíxel "j" tiene su propio conjunto de coeficientes de filtro.

20 En términos de simetría de coeficientes, los coeficientes de filtro para el primer grupo de subpíxeles que incluyen "a", "c", "d" y "l", usan un filtro monodimensional de 6 posiciones que tiene seis coeficientes únicos (es decir, siendo el primer grupo no simétrico), los coeficientes de filtro para el segundo grupo de subpíxeles "b" y "h" utilizan un filtro monodimensional de 6 posiciones que tiene tres coeficientes únicos, (es decir, b es simétrico), y un tercer conjunto de coeficientes de filtro para el grupo de subpíxeles "e", "g", "m" y "o" utiliza un filtro direccional de 6 posiciones que tiene seis coeficientes únicos. Un cuarto conjunto de coeficientes de filtro para el grupo "f", "i", "k" y "n" utiliza un filtro de 12 taps que tiene 6 coeficientes únicos y un quinto conjunto de coeficientes de filtro para el grupo "j" que tiene tres coeficientes únicos. El número total de coeficientes únicos en el esquema D-AIF es $6(a) + 3(b) + 6(e) + 6(f) + 3(j) = 24$ coeficientes. Estos coeficientes de filtro pueden predecirse, cuantificarse, codificarse y enviarse en el flujo de bits de vídeo.

25 La unidad MC 37 de la unidad de predicción 32 puede utilizar una o más de las técnicas de interpolación descritas en el presente documento para mejorar la precisión de la interpolación de posiciones de subpíxeles y/o reducir el número de coeficientes de filtro de interpolación que necesitan codificarse y transmitirse. A continuación se describe con más detalle un útil soporte de filtro de doce píxeles con una forma radial que se puede utilizar para la interpolación, técnicas que utilizan simetría de coeficientes y simetría de píxeles para reducir la cantidad de datos necesarios a enviar entre un codificador y un decodificador para configurar el soporte de filtro para la interpolación, técnicas para eliminar algunos aspectos de simetría en relación con las técnicas convencionales, y técnicas para filtrar datos en ubicaciones de píxeles enteros de una manera similar a la interpolación de subpíxeles. También se describen técnicas de desfase. Además, también se describen técnicas para seleccionar entre filtros de interpolación, y técnicas para codificar de forma predictiva coeficientes de filtro.

30 La FIG. 4 es un diagrama conceptual que ilustra un soporte de filtro de doce píxeles con respecto a nueve ubicaciones de subpíxeles. De acuerdo con esta descripción, la unidad MC 37 puede calcular valores de subpíxeles para las posiciones de subpíxeles "e", "f", "g", "i", "j", "k", "m", "n", y "o" en función de los valores de píxeles enteros sombreados. En este caso, el cálculo de los valores de subpíxeles comprende aplicar un filtro de interpolación que define una matriz bidimensional de posiciones de soporte de filtro correspondientes a un conjunto de doce (o posiblemente más) posiciones de píxeles enteros que rodean las posiciones de subpíxel como se ilustra en la FIG. 4. Las posiciones de soporte del filtro se muestran con sombreado y letras mayúsculas, y las posiciones de subpíxeles se muestran como cuadros con líneas discontinuas y letras minúsculas. La unidad MC 37 puede interpolar píxeles de esta manera para cada píxel de un bloque de vídeo, y puede generar un bloque de predicción basado en la interpolación de los valores de subpíxeles. El soporte de filtro de 12 posiciones tiene una forma radial y

tiene una respuesta de frecuencia similar al soporte de filtro de posición 36 utilizado en NS-AIF pero con una complejidad de interpolación mucho menor.

5 Como puede apreciarse a partir de la FIG. 3, los valores de subpíxeles pueden comprender quince valores de medio pel y cuarto de pel. El filtro de interpolación que define la matriz bidimensional de las posiciones de soporte de filtro se aplica para definir nueve de los quince valores de medio pel y cuarto de pel como se muestra en la FIG. 4.

10 El conjunto de doce o más posiciones de píxeles enteros que rodean las posiciones de subpíxeles se muestran con sombreado en la FIG. 4, y puede incluir cuatro posiciones de píxeles enteros C3, C4, D3 y D4 que rodean los nueve valores de medio pel y cuarto de pel, y ocho posiciones de píxeles enteros B3, B4, C2, C5, D2, D5, E3 y E4 que rodean las cuatro posiciones de píxeles enteros. Cada una de las posiciones de soporte del filtro en la matriz bidimensional de las posiciones de soporte del filtro mostradas en la FIG. 4 está dentro de dos posiciones de píxeles enteros con respecto a las posiciones de subpíxeles. Dicho filtrado puede formar parte de la compensación de movimiento durante un proceso de codificación de vídeo por el codificador de vídeo 50 (FIG. 2) o durante un proceso de decodificación de vídeo por el decodificador de vídeo 60 (FIG. 16). Durante el proceso de codificación de vídeo, la unidad de predicción 32 envía una pluralidad de conjuntos de valores de coeficientes a la unidad de codificación por entropía 46 como parte de los elementos de sintaxis de predicción. Como se muestra en la FIG. 4, el conjunto de doce o más posiciones de píxeles enteros que rodean las posiciones de subpíxeles tiene una forma radial y puede comprender un conjunto de posiciones de píxeles enteros espacialmente más cercanos a los nueve valores de medio pel y cuarto de pel definidos por el filtro de interpolación (por ejemplo, el filtro o filtros 39 en la FIG. 2).

25 La unidad de codificación por entropía 46 puede codificar los conjuntos de valores de coeficientes y emitir los valores de coeficiente del filtro de interpolación como parte de un flujo de bits codificado, que puede ser transmitido entonces a otro dispositivo. Cada una de la pluralidad de conjuntos de valores de coeficientes definidos con respecto a cada ubicación de subpíxel diferente mostrada en la FIG. 4 puede definir diferentes ponderaciones para las doce o más posiciones de píxeles enteros. La simetría de píxeles puede existir de tal forma que M conjuntos de valores de coeficientes definen N ponderaciones diferentes para las doce o más posiciones de píxeles enteros, en donde M y N son números enteros positivos y N es mayor que M. En otras palabras, la simetría de píxeles puede permitir que parte de las ubicaciones de píxeles se basen en ponderaciones reflejadas, invertidas o giradas de otras ubicaciones de píxeles de manera que los coeficientes no necesiten codificarse para ubicaciones que tienen simetría de píxeles con otras ubicaciones de subpíxeles.

35 La FIG. 5 es un diagrama conceptual que ilustra un soporte de filtro horizontal de seis píxeles con respecto a tres ubicaciones de subpíxeles horizontales, y un soporte de filtro vertical de seis píxeles con respecto a tres ubicaciones verticales de subpíxeles. Las ubicaciones de soporte de filtro están sombreadas con letras mayúsculas y las ubicaciones de subpíxeles no están sombreadas, tienen letras minúsculas y líneas discontinuas. La unidad MC 37 puede calcular los valores de subpíxeles para las ubicaciones "a", "b" y "c" aplicando un filtro de interpolación lineal que define un conjunto unidimensional de posiciones de soporte de filtro C1-C6. La unidad MC 37 también puede calcular los valores de subpíxeles para las ubicaciones "d", "h" y "l" aplicando un filtro de interpolación lineal que define una matriz unidimensional de posiciones de soporte de filtro A3-F3. De esta manera, se pueden usar filtros de interpolación lineal para seis de los quince valores de medio pel y cuarto de pel mostrados en la FIG. 3.

45 La FIG. 6 es un diagrama conceptual que ilustra un soporte de filtro de cinco píxeles por cinco píxeles para filtrar una ubicación de píxeles enteros. En este caso, se puede aplicar un filtrado tipo interpolación con respecto a las ubicaciones de píxeles enteros con el fin de generar valores de píxeles enteros ajustados. En la FIG. 6, por ejemplo, el valor de píxel entero en la posición C3 puede filtrarse en base a un conjunto de píxeles de cinco a cinco píxeles A1-A5, B1-B5, C1-C5, D1-D5 y E1-E5. Se pueden definir otros soportes de filtro para el filtrado de enteros. En este caso, si la unidad ME 35 identifica un bloque de píxeles enteros, la unidad MC 37 puede filtrar cada píxel en base a una matriz de píxeles de cinco por cinco mostrada en la FIG. 6 (u otro soporte de filtro) para generar valores de píxeles enteros filtrados. De esta manera, el filtrado de tipo interpolación puede aplicarse a píxeles enteros, lo que puede mejorar la codificación en momentos de cambios de iluminación o de transición o desvanecimiento de escena. Además, el filtrado de enteros puede eliminar el ruido y facilitar la nitidez del marco de la imagen, y puede ayudar a mejorar la codificación del movimiento del objeto fino entre tramas de vídeo sucesivas, particularmente cuando no se impone simetría en los coeficientes de filtro. El filtrado de pel enteros también puede ser extremadamente útil para mejorar la calidad y/o compresión de las secuencias de vídeo que tienen cambios de enfoque.

60 Por lo tanto, el codificador de vídeo 22 y/o el decodificador de vídeo 28 pueden utilizar el filtrado de tipo interpolación con respecto a posiciones de píxeles enteros con el fin de generar valores de píxeles enteros ajustados. El codificador de vídeo 22 o decodificador de vídeo 28 puede, por ejemplo, obtener bloques de píxeles, en el que los bloques de píxeles incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro del bloque de píxeles. El codificador de vídeo 22 o el decodificador de vídeo 28 puede filtrar los valores de píxeles enteros basados en otros valores de píxeles enteros dentro de los bloques de píxeles para generar valores de píxeles enteros ajustados, en el que los valores de píxeles enteros ajustados corresponden a las posiciones de píxeles enteros, y generar un bloque de predicción basado en los valores de píxeles enteros ajustados. El bloque de predicción puede usarse entonces para codificar o decodificar un bloque de vídeo, dependiendo de si el proceso se

realiza durante la codificación o decodificación.

Si la técnica se realiza durante un proceso de codificación, el codificador de vídeo 22 puede generar un bloque de vídeo residual como diferencia entre un bloque de vídeo actual que está siendo codificado y el bloque de predicción.

5 El codificador de vídeo 22 también puede codificar un conjunto de coeficientes de filtro por unidad codificada para filtrado de enteros o, posiblemente, codificar dos o más conjuntos de coeficientes de filtro por unidad codificada para filtrado entero en diferentes ubicaciones dentro de la unidad codificada. En el ejemplo de la FIG. 6, los soportes de filtro para el filtrado de números enteros pueden definir un filtro bidimensional, por ejemplo, un filtro de 5 por 5 con desfase. Además, puede existir simetría de coeficientes entre al menos algunos coeficientes utilizados para el filtrado de tal manera que sólo un subconjunto de los 25 coeficientes diferentes son necesarios para determinar los 25 coeficientes para el filtro. Pueden usarse muchos otros tamaños y formas de filtro de acuerdo con el filtrado de pel enteros.

15 Si se realiza el filtrado de pel enteros durante un proceso de decodificación, la técnica puede comprender generar un bloque de vídeo reconstruido como una suma de un bloque de vídeo residual asociado con un bloque de vídeo actual que está siendo decodificado y el bloque de predicción. En este caso, el dispositivo de decodificación (por ejemplo, el dispositivo de destino 16) puede recibir un conjunto de coeficientes de filtro por unidad codificada para filtrar números enteros, o puede recibir dos o más conjuntos de coeficientes de filtro por unidad codificada para filtrado de enteros en diferentes ubicaciones dentro de la unidad codificada. Cada unidad codificada puede tener uno o más conjuntos de coeficientes de filtrado de pel enteros, así como conjuntos de coeficientes de filtración de subpel, por ejemplo, ocho conjuntos que se pueden usar para generar los quince conjuntos para posiciones de medio pel y cuarto de pel.

25 De nuevo, el filtrado de interpolación de valores de píxeles enteros puede mejorar la compresión específicamente durante los cambios de iluminación, la transición o desvanecimiento de escena, puede eliminar el ruido y facilitar la nitidez del marco de la imagen, y puede ayudar a mejorar la codificación del movimiento del objeto fino entre tramas de vídeo sucesivas, particularmente cuando no se impone simetría en los coeficientes de filtro. Además, el filtrado de pel enteros también es extremadamente útil, por ejemplo, para mejorar la compresión, en secuencias de vídeo que tienen cambios de enfoque.

30 Convencionalmente, el filtrado de tipo interpolación no se realiza para posiciones de píxeles enteros en AIF. En su lugar, la posición correspondiente de píxeles enteros en la trama de referencia se utiliza simplemente "tal cual". Sin embargo, de acuerdo con esta divulgación, se puede usar un filtro para ubicaciones de píxeles enteros de una manera que es similar a las interpolaciones. El filtrado de interpolación en posiciones de píxeles enteros es técnicamente un nombre incorrecto, ya que los valores de píxeles de posiciones de píxeles enteros ya existen. Por lo tanto, el filtrado de posiciones de píxeles enteros se denomina en el presente documento filtrado de tipo interpolación, y se puede ver simplemente filtrando los valores de píxeles enteros para generar nuevos valores de píxeles ajustados.

40 A diferencia del filtrado de interpolación para posiciones de subpíxeles, donde el objetivo es obtener los valores inexistentes en la trama de referencia, los píxeles en las posiciones enteras en la trama de referencia ya existen. Sin embargo, la aplicación de filtrado en las posiciones de píxeles enteros en la trama de referencia ofrece una serie de beneficios. Por ejemplo, cuando las posiciones de píxeles enteros contienen un desfase, el filtrado de las posiciones de píxeles enteros puede permitir una mejor captura de los cambios de iluminación entre tramas. Además, el filtrado en las posiciones de píxeles enteros puede permitir que las posiciones de píxeles enteros capturen efectos especiales tales como transición y desvanecimiento. El filtrado en las posiciones de píxeles enteros también puede eliminar ciertos ruidos de señal (como el ruido de la cámara) y/o realizar la nitidez, si es necesario. Además, si no se impone ninguna simetría en el filtro de píxeles enteros, el filtro de píxeles enteros puede ayudar a capturar el movimiento del objeto fino, como el movimiento del objeto que no es un múltiplo de un cuarto de píxel. Por último, el filtrado de píxeles enteros también puede ser útil para mejorar la compresión de secuencias de vídeo cuando se producen cambios de enfoque.

55 En algunos casos, el filtrado de interpolación de posiciones de píxeles enteros puede depender de vectores de movimiento. En otras palabras, la unidad MC 37 de la unidad de predicción 32 puede aplicar de forma adaptativa el filtrado de interpolación basado en el vector de movimiento identificado por la unidad ME 35. Por ejemplo, el filtrado de interpolación de posiciones de píxeles enteros se puede realizar cuando el vector de movimiento apunta a una posición de píxel entero. El filtrado de interpolación puede dar como resultado un valor de píxel ajustado para la posición de píxeles enteros. Cuando el vector de movimiento apunta a una posición de subpíxel, no se realiza ningún filtrado de posiciones de píxeles enteros con respecto a los valores de píxeles enteros, pero puede utilizarse el filtrado para la interpolación de los valores de subpíxeles. De esta manera, el filtrado de píxeles enteros puede considerarse adaptativo basado en vectores de movimiento.

60 Para las posiciones de píxeles enteros, se puede derivar un filtro de interpolación de 5 por 5 en la forma de la ecuación (15) usando el mismo proceso analítico usado para derivar los filtros de posición de subpíxeles.

65

$$p^{FP} = \sum_{i=0}^4 \sum_{j=0}^4 P_{i,j} h_{i,j}^{FP} + o^{FP}, \quad (15)$$

donde p^{FP} es el valor de píxel entero filtrado, $P_{i,j}$ es el valor de píxel entero en la posición (i, j) , $h_{i,j}^{FP}$ es el coeficiente para la posición (i, j) , y o^{FP} es un desfase de filtro. En algunos casos, el filtro de interpolación para posiciones de píxeles enteros puede no tener un desfase de filtro (es decir, $o^{FP} = 0$). En otros casos, el desfase de filtro o^{FP} puede utilizar cualquiera de una serie de técnicas.

Pueden definirse diferentes desfases para cada posición de subpíxel, por ejemplo, pueden definirse quince distancias diferentes para cada posición de subpíxel mostrada en la FIG. 3. Se puede definir otro desfase para píxeles enteros, llevando el número total de desplazamientos a dieciséis. La Solicitud de Patente de Estados Unidos pendiente junto con la presente y comúnmente asignada n.º de serie _____, presentada por Karczewicz et al., y titulada "OFFSETS AT SUBPIXEL RESOLUTION" con el número de registro 081399U2, que se presentó el mismo día que la presente solicitud, y que se incorpora en el presente documento por referencia, proporciona muchos detalles adicionales sobre el uso de muchos desfases para diferentes ubicaciones de píxeles y subpíxeles.

Con el fin de reducir el número de coeficientes de filtro a transmitir, la simetría de coeficientes como en la ecuación (16) puede imponerse al filtro de píxeles enteros.

$$h_{i,j}^{FP} = h_{i,4-j}^{FP} = h_{4-i,j}^{FP} = h_{4-i,4-j}^{FP}, \text{ for } i, j = 0..4 \quad (16)$$

También pueden usarse otras formas de simetría de coeficientes. Si la simetría de coeficiente se impone puede ser decidido por el codificador y señalado al decodificador. Si se utiliza la simetría como en (16) en el filtro de píxeles enteros, entonces puede ser necesario enviar 9 coeficientes de filtro y 1 desfase al decodificador.

La codificación de los desfases puede hacerse codificando primero un desfase entero y luego codificando cada desplazamiento fraccionario. En este caso, el desfase de enteros y los desfases fraccionarios se codifican de manera diferente. El desfase de enteros puede codificarse usando código Golomb exponencial firmado. Dependiendo de la magnitud del desfase de enteros, los desfases fraccionarios pueden cuantificarse de manera diferente que el desfase de enteros. Por ejemplo, pueden aplicarse niveles más gruesos de cuantificación en los desplazamientos fraccionarios cuando un desplazamiento entero tiene una gran magnitud. Después de la cuantificación, los desfases fraccionarios pueden codificarse usando códigos de longitud fija. Como ejemplo, el siguiente pseudo-código puede ilustrar cómo se puede codificar un determinado desplazamiento fraccionario usando diferentes códigos de longitud fija definidos en base al valor de desfase entero.

```

Se espera que offsetI sea el desfase de enteros
Se espera que offsetF sea el desfase fraccional
Se espera que offsetFbits sea el número de bits usado para cuantificar offsetF
Se espera que abs(a) sea el valor absoluto de
    if(abs(offsetI) = 0)
        offsetFbits = 5
    else if(abs(offsetI) < 3)
        offsetFbits = 4
    else if(abs(offsetI) < 7)
        offsetFbits = 2
    else
        offsetFbits = 0
    
```

En este caso, el codificador de vídeo 50 puede usar una unidad de predicción y cuantificación de coeficientes 41 que para cada uno de los valores de desplazamiento asigna un primer número de bits a una porción entera de un valor de desfase dado y asigna un segundo número de bits a una porción fraccionada del valor de desfase dado, en el que el primer y segundo números de bits se determinan basándose en una magnitud de la porción entera. En este caso, las versiones cuantificadas de los valores de desfase pueden enviarse desde la unidad de predicción 32 como elementos de sintaxis de predicción después de que la cuantificación de los valores de desplazamiento se lleve a cabo de forma consistente con el pseudo-código anterior. En cualquier caso, el codificador de vídeo 50 puede codificar el primer número de bits de manera diferente que el segundo número de bits.

La FIG. 7 es un diagrama conceptual que ilustra cuatro posiciones de píxeles enteros y quince posiciones de subpíxeles con sombreado para posiciones de píxeles en grupos que pueden utilizar simetría de píxeles para coeficientes de filtro coherentes con esta divulgación. En particular, la FIG. 7 ilustra las posiciones de subpíxel "a" a "o" correspondientes a la posición de píxel entero C3. Las posiciones de subpíxeles enteros mostradas en la FIG. 7 corresponden a las de las FIGS. 4 y 5. Es decir, las ubicaciones de subpíxeles "e", "f", "g", "i", "j", "k", "m", "n", y "o" pueden tener valores determinados por un soporte de doce píxeles como se muestra en la FIG. 4, mientras que las

ubicaciones de subpíxeles "a", "b" y "c" y las ubicaciones de subpíxeles "d", "h" y "i" pueden tener valores determinados por soportes de píxeles horizontales y lineales mostrados en la FIG. 5.

5 En este escenario, la simetría de píxeles puede imponerse de tal manera que los coeficientes de filtro para la ubicación de píxeles "a" sean simétricos con respecto a los coeficientes de filtro para la ubicación de píxeles "c". De forma similar, la simetría de píxeles puede imponerse de tal manera que los coeficientes de filtro para la ubicación de píxeles "d" sean simétricos con respecto a los coeficientes de filtro para la ubicación de píxeles "i". La simetría de píxeles puede imponerse de tal manera que los coeficientes de filtro para la ubicación de píxeles "e" sean simétricos con respecto a los coeficientes de filtro para ubicaciones de píxeles "g", "m" y "o", y la simetría de píxeles puede imponerse de tal manera que los coeficientes de filtro para la ubicación de píxeles "i" son simétricos con respecto a los coeficientes de filtro para ubicaciones de píxeles "k", y la simetría de píxeles puede imponerse de tal manera que los coeficientes de filtro para la ubicación de píxeles "f" sean simétricos con respecto a los coeficientes de filtro para ubicaciones de píxeles "n". Por lo tanto, sólo se necesitan comunicar ocho conjuntos de valores de coeficientes como parte de un flujo de bits de vídeo para permitir que un decodificador genere conjuntos completos de valores de coeficiente para las 15 ubicaciones de píxeles. Además, para cualquier conjunto dado de coeficientes, la simetría de coeficiente puede permitir que solamente se envíe un subconjunto de los valores de coeficiente, y el decodificador puede generar el conjunto completo de valores de coeficiente para una ubicación de píxel dada en base al subconjunto y la simetría de coeficientes.

20 Un aspecto de esta divulgación es eliminar realmente cualquier simetría de píxeles entre las posiciones de subpíxeles "a" y "c" en relación con las posiciones de subpíxeles "d" y "i". Además, se puede evitar la simetría de píxeles para la posición de subpíxeles "f" con respecto a la posición de subpíxel "i". Además, se puede evitar la simetría de píxeles para la posición de subpíxel "k" con relación a la posición de subpíxel "n". En tales casos, la correlación diagonal puede ser baja, lo que hace ineficiente o ineficaz para imponer simetría diagonal. De esta manera, los ocho conjuntos de coeficientes utilizados para generar los quince conjuntos de coeficientes pueden ser un conjunto mayor que algunas técnicas convencionales en la medida en que se evita la simetría en la dimensión diagonal para algunas de las ubicaciones de los píxeles. En algunos casos, la simetría de píxeles diagonales y la simetría de coeficientes diagonales pueden eliminarse o evitarse para permitir la variación en las dimensiones verticales.

30 Con referencia de nuevo a la FIG. 1, el codificador de vídeo 22 puede determinar ocho conjuntos de coeficientes de filtro y utilizar los ocho conjuntos de coeficientes de filtro para generar los quince conjuntos de coeficientes de filtro correspondientes a quince ubicaciones de subpíxeles diferentes basadas en la simetría de coeficiente y simetría de píxeles, y emitir los ocho conjuntos de coeficientes de filtro como parte de un flujo de bits codificado. Pueden emitirse ocho conjuntos de coeficientes de filtro para cada unidad codificada del flujo de bits codificado, o posiblemente se pueden producir varias apariciones de los ocho conjuntos de coeficientes de filtro para diferentes áreas de una unidad codificada.

40 Los quince conjuntos corresponden a cada ubicación de subpíxel de media pel y cuarto de pel mostrada en la FIG. 3, y los ocho conjuntos, la simetría de coeficientes y la simetría de píxeles definen soportes de filtro para cada ubicación de subpíxel de medio pel y cuarto de pel. El decodificador de vídeo 28 del dispositivo de destino 16 puede recibir ocho conjuntos de coeficientes de filtro como parte de un flujo de bits de vídeo codificado, generar quince conjuntos de coeficientes de filtro correspondientes a quince ubicaciones de subpíxeles diferentes basadas en los ocho conjuntos de coeficientes de filtro, generar datos predictivos interpolados para decodificación de vídeo en base a uno de los quince conjuntos de coeficientes de filtro, en el que los datos predictivos interpolados corresponden a una de las quince ubicaciones de subpíxeles diferentes, y decodificar uno o más bloques de vídeo en base a los datos predictivos interpolados.

50 Al menos algunos de los ocho conjuntos de coeficientes de filtro pueden tener una simetría de coeficientes de tal forma que un conjunto dado con simetría de coeficientes comprende un subconjunto de los coeficientes de filtro asociados con un soporte de filtro para una ubicación de subpíxeles dada asociada con el conjunto dado. De este modo, el decodificador de vídeo 28 puede generar los quince conjuntos de coeficientes de filtro basados al menos en parte en la simetría de píxeles entre dos o más ubicaciones de subpíxeles diferentes. Además, el decodificador de vídeo 28 puede generar los quince conjuntos de coeficientes de filtro basados al menos en parte en la simetría de píxeles entre dos o más ubicaciones de subpíxeles diferentes, y en base, al menos en parte, a la simetría de coeficientes definida para un conjunto dado de modo que el conjunto dado con simetría de coeficientes comprende un subconjunto de los coeficientes de filtro asociados con un soporte de filtro para una ubicación de subpíxeles dada asociada con el conjunto dado.

60 De nuevo, los quince conjuntos corresponden a cada ubicación de subpíxel de media pel y cuarto de pel. La simetría de píxeles puede existir entre diferentes de los quince conjuntos de coeficientes de filtro en una dimensión vertical y en una dimensión horizontal, pero la simetría de píxeles no existe en una dimensión diagonal para al menos algunos de los quince conjuntos de coeficientes de filtro. Como se explica con mayor detalle a continuación, dos de los ocho conjuntos pueden definir tres de los quince conjuntos para un soporte de filtro horizontal, dos de los ocho conjuntos pueden definir tres de los quince conjuntos para un soporte de filtro vertical, y cuatro de los ocho conjuntos pueden definir nueve de los quince conjuntos para un soporte de filtro bidimensional. El soporte bidimensional puede

comprender un soporte de filtro de 12 posiciones útil que rodea nueve posiciones de subpíxeles en una forma radial.

Específicamente, los datos predictivos interpolados pueden comprender quince valores de medio pel y cuarto de pel, por ejemplo, mostrados en la FIG. 7, en los que el filtro de interpolación define una matriz bidimensional de doce posiciones de soporte de filtro, por ejemplo, mostradas en la FIG. 4, que se aplican para definir nueve de los quince valores de medio pel y cuarto de pel y, por ejemplo, también mostrados en la FIG. 4. La generación de los datos predictivos interpolados puede comprender además la aplicación de filtros de interpolación lineal que definen conjuntos unidimensionales de posiciones de soporte de filtro para seis de los quince valores de medio pel y cuarto de pel, por ejemplo, como se muestra en la FIG. 5.

La simetría horizontal y la falta de simetría diagonal se puede observar con respecto a las posiciones de subpíxeles "a" y "c" o con respecto a las posiciones de subpíxeles "i" y "k". En este caso, las posiciones de subpíxeles enteros "a" y "c" son simétricas a lo largo de un eje X horizontal, pero la simetría no existe a lo largo de un eje diagonal para estos píxeles de modo que las ubicaciones de subpíxeles "d" y "l" no son simétricas con respecto a las posiciones de subpíxeles "a" y "c" respectivamente. De forma similar, las posiciones de subpíxeles "i" y "k" son simétricas a lo largo de un eje X horizontal, pero no existe simetría a lo largo del eje diagonal para estos píxeles de modo que las ubicaciones de subpíxeles "f" y "n" no son simétricas con respecto a las posiciones de subpíxeles "i" y "k" respectivamente.

La simetría de píxeles significa que un primer conjunto de coeficientes con respecto a la primera posición de subpíxel es simétrica con respecto a un segundo conjunto de coeficientes con respecto a la segunda posición de subpíxel. Por ejemplo, la ubicación de píxel "a" tiene simetría de píxeles con respecto a la ubicación de píxel "c", y la ubicación de píxel "d" tiene simetría de píxeles con respecto a la ubicación de píxel "l". La ubicación de píxel "f" tiene simetría de píxeles con respecto a la ubicación de píxel "n", y la ubicación de píxel "i" tiene simetría de píxeles con respecto a la ubicación de píxel "k". La ubicación de píxel "e" tiene simetría de píxeles con respecto a las ubicaciones de píxeles "g", "m" y "o". El sombreado mostrado en la FIG. 7 muestra esta simetría de píxeles, por ejemplo, con sombreado común correspondiente a ubicaciones de píxeles que son simétricas con respecto a otras ubicaciones de píxeles. En este caso, los coeficientes para ocho ubicaciones de píxeles (junto con aspectos de simetría de píxeles) pueden definir coeficientes para las 15 ubicaciones de píxeles. Además, dentro de un conjunto dado de coeficientes, puede existir una simetría de coeficientes de tal forma que sólo un subconjunto de los coeficientes de ese conjunto de coeficientes necesita comunicarse con el flujo de bits.

El codificador de vídeo 22 puede generar datos de vídeo codificados basados en datos predictivos interpolados, en el que los datos predictivos interpolados comprenden quince valores de medio pel y cuarto de pel, en el que el filtro de interpolación define una matriz bidimensional de doce posiciones de soporte de filtro que se aplican para definir nueve de los quince valores de medio pel y cuarto de pel. La generación de los datos de vídeo codificados basados en datos predictivos interpolados puede comprender además la aplicación de filtros de interpolación lineal que definen conjuntos unidimensionales de posiciones de soporte de filtro para seis de los quince valores de medio pel y cuarto de pel.

La FIG. 8 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales horizontales C1-C6 con relación a una posición de subpíxel "b", con sombreado que muestra simetría de coeficientes. En este caso, la simetría de coeficientes significa que sólo se necesitan tres coeficientes de filtro para C1, C2 y C3 para definir el conjunto completo de coeficientes para las posiciones de soporte de filtro C1-C6. C1 es simétrica con C6, C2 es simétrica con C5 y C3 es simétrica con C4. Por lo tanto, sólo se deben comunicar tres coeficientes como parte de un flujo de bits de vídeo codificado para definir el conjunto de seis coeficientes necesarios para interpolar la posición de subpíxel "b".

La FIG. 9 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales horizontales con relación a un subpíxel, con sombreado que muestra una falta de simetría de coeficientes. Por lo tanto, los seis coeficientes son necesarios para definir el conjunto de coeficientes para el soporte del filtro con respecto a la posición de subpíxel "a". Sin embargo, como se ha indicado anteriormente, la simetría de píxeles significa que estos mismos coeficientes con respecto a la posición de subpíxel "a" también se pueden usar para derivar el soporte de filtro para la posición de subpíxel "c" (véase la FIG. 7). Por lo tanto, sólo se deben comunicar seis coeficientes como parte de un flujo de bits de vídeo codificado para definir dos conjuntos diferentes de seis coeficientes necesarios para interpolar la posición de subpíxel "a" y "c".

La FIG. 10 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales verticales A3, B3, C3, D3, E3 y F3 con relación a un subpíxel "h", con sombreado que muestra simetría de coeficientes. En este caso, la simetría de coeficientes significa que sólo se necesitan tres coeficientes de filtro para A3, B3 y C3 para definir el conjunto completo de coeficientes para las posiciones de soporte de filtro A3, B3, C3, D3, E3 y F3. A3 es simétrica con F3, B3 es simétrica con E3 y C3 es simétrica con D3. Por lo tanto, sólo se deben comunicar tres coeficientes como parte de un flujo de bits de vídeo codificado para definir el conjunto de seis coeficientes necesarios para interpolar la posición de subpíxel "h".

La FIG. 11 es un diagrama conceptual que ilustra seis posiciones de soporte de píxeles lineales verticales con

relación a un subpíxel, con sombreado que muestra una falta de simetría de coeficientes.

Por lo tanto, los seis coeficientes son necesarios para definir el conjunto de coeficientes para el soporte del filtro con respecto a la posición de subpíxel "d". Sin embargo, como se ha indicado anteriormente, la simetría de píxeles significa que estos mismos coeficientes con respecto a la posición de subpíxel "d" también se pueden usar para derivar el soporte de filtro para la posición de subpíxel "l" (véase la FIG. 7). Por lo tanto, sólo se deben comunicar seis coeficientes como parte de un flujo de bits de vídeo codificado para definir dos conjuntos diferentes de seis coeficientes necesarios para interpolar la posición de subpíxel "d" y "l".

La FIG. 12 es un diagrama conceptual que ilustra doce posiciones de soporte de píxeles bidimensionales (las posiciones de píxeles enteros mostradas con letras mayúsculas y sombreado) con relación a una posición de subpíxel "e". El sombreado muestra una falta de cualquier simetría de coeficientes. Por lo tanto, los doce coeficientes son necesarios para definir el conjunto de coeficientes para el soporte del filtro con respecto a la posición de subpíxel "e". Sin embargo, como se ha indicado anteriormente, la simetría de píxeles significa que estos mismos coeficientes con respecto a la posición de subpíxel "e" también se pueden usar para derivar el soporte de filtro para las posiciones de subpíxeles "g", "m" y "o" (véase la FIG. 7). Por lo tanto, 12 coeficientes definen la posición de subpíxel "e" de soporte de filtro y estos mismos coeficientes pueden utilizarse para derivar el conjunto de coeficientes para posiciones de subpíxel "g", "m" y "o".

La FIG. 13 es un diagrama conceptual que ilustra doce posiciones de soporte de píxeles bidimensionales (las posiciones de píxeles enteros mostradas con letras mayúsculas) con relación a un subpíxel "e", con un sombreado que muestra simetría de coeficientes. En este caso, la simetría de coeficientes significa que los coeficientes de filtro para las posiciones de soporte de filtro B3, B4, C2, C3, C4 y C5 pueden usarse para definir los coeficientes de filtro para las posiciones de soporte de filtro E3, E4, D2, D3, D4 y D5. B3 es simétrica con E3, B4 es simétrica con E4, C2 es simétrica con D2, C3 es simétrica con D3, C4 es simétrica con D4 y C5 es simétrica con D5. Por lo tanto, sólo se deben comunicar seis coeficientes como parte de un flujo de bits de vídeo codificado para definir el conjunto de doce coeficientes necesarios para interpolar la posición de subpíxel "i". Además, como se ha indicado anteriormente, la posición de subpíxel "i" puede tener simetría de píxeles con respecto a la posición de subpíxel "k". Por lo tanto, el mismo subconjunto de seis coeficientes puede definir los conjuntos completos de doce coeficientes de filtro para las posiciones de subpíxeles "i" y "k".

La FIG. 14 es un diagrama conceptual que ilustra doce posiciones de soporte de píxeles bidimensionales (las posiciones de píxeles enteros mostradas con letras mayúsculas) con relación a un subpíxel "f", con un sombreado que muestra simetría de coeficientes. En este caso, la simetría de coeficientes significa que los coeficientes de filtro para las posiciones de soporte de filtro C2, D2, B3, C3, D3 y E3 pueden usarse para definir los coeficientes de filtro para las posiciones de soporte de filtro C5, D5, B4, C4, D4 y E4. C2 es simétrica con C5, D2 es simétrica con D5, B3 es simétrica con B4, C3 es simétrica con C4, D3 simétrica con D4 y E3 es simétrica con E4. Por lo tanto, sólo se deben comunicar seis coeficientes como parte de un flujo de bits de vídeo codificado para definir el conjunto de doce coeficientes necesarios para interpolar la posición de subpíxel "f". Además, como se ha indicado anteriormente, la posición de subpíxel "f" puede tener simetría de píxeles con respecto a la posición de subpíxel "n". Por lo tanto, el mismo subconjunto de seis coeficientes puede definir los conjuntos completos de doce coeficientes de filtro para las posiciones de subpíxeles "f" y "n".

La FIG. 15 es un diagrama conceptual que ilustra doce posiciones de soporte de píxeles bidimensionales (las posiciones de píxeles enteros mostradas con letras mayúsculas) con relación a un subpíxel "j", con un sombreado que muestra simetría de coeficientes. En este caso, la simetría de coeficientes significa que sólo se necesitan tres coeficientes de filtro para B3, C2 y C3 para definir el conjunto completo de doce coeficientes para las posiciones de soporte de filtro B3, B4, C2, C3, C4, C5, D2, D3, D4, D5, E3 y E4. C3 es simétrica con C4, D3 y D4, C2 es simetría con D2, C5 y E4, y B3 es simétrica con B4, E3 y D5. Por lo tanto, sólo se deben comunicar tres coeficientes como parte de un flujo de bits de vídeo codificado para definir el conjunto de doce coeficientes necesarios para interpolar la posición de subpíxel "j".

La FIG. 16 es un diagrama de bloques que ilustra un ejemplo de un decodificador de vídeo, que puede decodificar una secuencia de vídeo que se codifica de manera descrita en el presente documento. El decodificador de vídeo 60 es un ejemplo de dispositivo o aparato informático de vídeo especializado, denominado "codificador" en el presente documento. El decodificador de vídeo 60 incluye una unidad de decodificación por entropía 52 que realiza la decodificación por entropía del flujo de bits recibido para generar coeficientes cuantificados y los elementos sintácticos de predicción. Los elementos sintácticos de predicción pueden incluir un modo codificación, uno o más vectores de movimiento, información que identifica una técnica de interpolación utilizada para generar los datos de subpíxeles, coeficientes para utilizar en el filtrado de interpolación y otra información asociada con la generación del bloque de predicción.

Los elementos sintácticos de predicción, por ejemplo, los coeficientes, se transmiten a la unidad de predicción 55. Si se utiliza la predicción para codificar los coeficientes en relación con los coeficientes de un filtro fijo, o en relación unos con otros, la unidad de predicción de coeficientes y cuantificación inversa 53 puede decodificar los elementos sintácticos para definir los coeficientes reales. Además, si se aplica la cuantificación a cualquiera de la sintaxis de

predicción, la unidad de predicción de coeficientes y de cuantificación inversa 53 puede también eliminar dicha cuantificación. Los coeficientes de filtro, por ejemplo, pueden codificarse mediante predicción y cuantificarse de acuerdo con la presente divulgación, y en este caso, la unidad de predicción de coeficientes y cuantificación inversa 60 puede utilizarse por la unidad de cuantificación inversa 53 para decodificar mediante predicción y decuantificar dichos coeficientes.

La unidad de predicción 55 puede generar datos de predicción basándose en los elementos sintácticos de predicción y uno o más bloques previamente decodificados que se almacenan en la memoria 62, de la misma manera que se ha descrito con detalle anteriormente con respecto a la unidad de predicción 32 del codificador de vídeo 50. En particular, la unidad de predicción 55 puede realizar una o más de las técnicas de filtrado de interpolación de esta divulgación durante la compensación de movimiento para generar un bloque de predicción con una precisión particular, tal como precisión de un cuarto de píxel. Así pues, el decodificador de vídeo 60 puede utilizar una o más de las técnicas de la presente divulgación en la generación de un bloque de predicción. La unidad de predicción 55 puede incluir una unidad de compensación de movimiento que comprende filtros utilizados para técnicas de filtración de interpolación y similares a la interpolación según la presente divulgación. El componente de compensación de movimiento no se muestra en la FIG. 16 con fines de simplicidad y facilidad de ilustración.

La unidad de cuantificación inversa 56 realiza la cuantificación inversa, es decir, decuantifica, los coeficientes cuantificados. El proceso de cuantificación inversa puede ser un proceso definido por la decodificación H.264. La unidad de transformada inversa 58 aplica una transformada inversa, por ejemplo una DCT inversa o un proceso de transformada inversa conceptualmente similar, a los coeficientes de transformada, con el fin de generar bloques residuales en el dominio del píxel. El sumador 64 suma el bloque residual con el correspondiente bloque de predicción generado por la unidad de predicción 55 para formar una versión reconstruida del bloque original codificado por el codificador de vídeo 50. Si se desea, también puede aplicarse un filtro de desbloqueo para filtrar los bloques descodificados con el fin de eliminar distorsiones de efecto pixelado. Los bloques de vídeo decodificados se almacenan a continuación en el almacenamiento de trama de referencia 62, que proporciona bloques de referencia para la compensación de movimiento posterior y también proporciona vídeo decodificado para activar el dispositivo de visualización (tal como el dispositivo 30 de la FIG. 1).

Las técnicas NS-AIF pueden calcular valores de píxeles para las posiciones de subpíxeles bidimensionales utilizando un filtro de interpolación que tiene soporte de filtro de 36 posiciones (es decir, un soporte de filtro bidimensional 6x6). S-AIF utiliza posiciones de píxeles enteros separables como soporte de filtro para el filtrado de interpolación primero en una dirección horizontal en lugar de en una dimensión vertical. D-AIF utiliza un filtro de interpolación menos complejo para calcular las posiciones de subpíxeles bidimensionales que NS-AIF. En D-AIF, las posiciones de subpíxeles bidimensionales se calculan utilizando un filtro de interpolación que tiene un soporte de filtro de 6 posiciones o un soporte de filtro diagonal de 12 posiciones.

Un inconveniente de utilizar el soporte de filtro diagonal como en D-AIF es que las posiciones de píxeles enteros usadas en el filtrado están muy lejos de la posición actual a interpolar. A medida que aumenta la distancia entre la posición actual a interpolar y las posiciones de los píxeles utilizados como soporte de filtro, disminuye la correlación espacial de píxeles. Como tal, el soporte de filtro diagonal usado en D-AIF es menos adecuado para formar una predicción exacta.

Con el fin de proporcionar una mejor predicción utilizando píxeles con mayor correlación con la posición a interpolar (es decir, posiciones de píxeles que están más cerca o una distancia más corta de la posición a interpolar), manteniendo la misma baja complejidad ofrecida por D-AIF, el soporte de filtro de 12 posiciones mostrado en la FIG. 4 puede usarse para la interpolación. Las posiciones de píxeles enteros utilizadas en el soporte de filtro de 12 posiciones descrito en esta divulgación incluyen las cuatro posiciones de píxeles enteros que rodean las posiciones de subpíxeles, es decir, las posiciones C3, C4, D3 y D4 de píxeles enteros, que pueden denominarse como posiciones de píxeles enteros de "esquina" cuando se sitúan cerca de las esquinas de las posiciones de subpíxeles bidimensionales. Además de las posiciones de píxeles enteros de esquina, también se pueden usar en el soporte de filtro una o más posiciones de píxeles enteros inmediatamente adyacentes a las posiciones de píxeles enteros de esquina. Las posiciones de píxeles enteros inmediatamente adyacentes a las posiciones de píxeles enteros de esquina pueden incluir posiciones de píxeles enteros directamente por encima (hacia la parte superior o norte) o directamente por debajo (hacia la parte inferior o sur) de las posiciones de píxeles enteros de esquina, así como posiciones de píxeles enteros directamente a la izquierda (o al oeste) o directamente a la derecha (o al este) de la esquina de las posiciones de píxeles enteros. En el ejemplo ilustrado en la FIG. 4, el soporte de filtro de 12 posiciones puede incluir ubicaciones de píxeles enteros B3, B4, C2, C3, C4, C5, D2, D3, D4, D5, E3 y E4. Todas estas posiciones de píxeles enteros están situadas dentro de dos posiciones de píxeles enteros del subpíxel a interpolar.

El soporte de filtro de acuerdo con las técnicas de esta divulgación puede, sin embargo, incluir ubicaciones adicionales de píxeles enteros. Por ejemplo, el soporte de filtro puede incluir ubicaciones de píxeles enteros que están situadas dentro de tres posiciones de píxeles enteros de la posición de subpíxel a interpolar que forman una forma radial.

Incluso las posiciones de un cuarto de píxel o una resolución más fina pueden interpolarse basándose en posiciones de píxeles enteros situadas dentro de al menos dos posiciones de píxeles enteros. De esta manera, las técnicas de soporte de filtro de esta divulgación proporcionan una mejor predicción de interpolación usando píxeles enteros con mayor correlación con la posición a interpolar, manteniendo al mismo tiempo la misma baja complejidad ofrecida por D-AIF.

Como se ha descrito anteriormente, la simetría de subpíxeles puede reforzar el mismo conjunto de coeficientes de filtro (y las versiones reflejadas, invertidas y/o giradas del conjunto) para un grupo de una o más posiciones de subpíxeles. Utilizando un proceso analítico, tal como el de la ecuación (5), se pueden usar coeficientes de filtro de interpolación para todas las posiciones de subpíxeles, por ejemplo, quince posiciones de subpíxeles en el caso de una precisión de un cuarto de píxel. El análisis de correlación entre los quince conjuntos de coeficientes de filtro de interpolación revela una correlación relativamente fuerte entre los coeficientes de filtro de las posiciones de subpíxeles (ambas posiciones de subpíxeles monodimensionales y bidimensionales) en la misma dimensión (por ejemplo, dimensión vertical u horizontal). Por el contrario, la correlación entre los coeficientes de filtro de las posiciones de subpíxeles en diferentes direcciones diagonales puede ser notablemente más débil.

Por ejemplo, puede haber una alta correlación entre los coeficientes de filtro para las posiciones de subpíxel "a" y "c", que están ambas en la dimensión horizontal. Como otro ejemplo, puede haber una alta correlación entre los coeficientes de filtro para las posiciones de subpíxel "d" y "l", que están ambas en la dimensión vertical. Sin embargo, puede haber poca o ninguna correlación entre los coeficientes de filtro para las posiciones de subpíxel "a", que está en la dimensión horizontal, y la posición de subpíxel "d", que está en la dimensión vertical. Basándose en estas observaciones de las correlaciones, la simetría de subpíxeles puede diseñarse de manera que no se imponga una simetría de subpíxeles para coeficientes de filtro de posiciones de subpíxeles en dimensiones diagonales. Esto conduce a ocho conjuntos de coeficientes, como se describe en el presente documento, que pueden ser más conjuntos de los necesarios para algunos procedimientos convencionales. Al eliminar la simetría de píxeles diagonales para algunas ubicaciones de píxeles, tal como se describe en el presente documento, la codificación y la compresión de vídeo pueden mejorarse.

Una vez más, la FIG. 7 muestra las posiciones de los píxeles con los mismos sombreados que pertenecen a un grupo de posiciones de subpíxeles que tienen simetría subpíxeles. En particular, las posiciones de subpíxeles "a" y "c" forman un primer grupo que tiene simetría de subpíxeles, las posiciones de subpíxeles "d" y "l" forman un segundo grupo que tiene simetría de subpíxeles, las posiciones de subpíxeles "e", "g", "m", y "o" forman un tercer grupo que tiene simetría de subpíxeles, las posiciones de subpíxeles "f" y "n" forman un cuarto grupo que tiene simetría de subpíxeles, y las posiciones de subpíxeles "i" y "k" forman un quinto grupo que tiene simetría de subpíxeles. Las posiciones de subpíxeles "b", "h" y "j" no tienen simetría de subpíxeles con ninguna otra posición de subpíxel. Como tal, las posiciones de subpíxeles "b", "h" y "j" pueden considerarse como pertenecientes a sus propios grupos; un sexto, séptimo y octavo grupo, respectivamente. Cada grupo de coeficientes (que pueden ser subconjuntos debido a la simetría de coeficientes) puede comunicarse una vez por unidad codificada, o posiblemente múltiples veces por unidad codificada si se definen diferentes tipos de filtrado para diferentes áreas o características de una unidad codificada. Los coeficientes para el filtrado de tipo interpolación de píxel enteros también se pueden enviar una o varias veces por unidad codificada.

En NS-AIF y D-AIF existen simetrías de subpíxeles entre posiciones de subpíxeles en la misma dimensión, por ejemplo, dos posiciones de subpíxeles en la dimensión vertical o dos posiciones de subpíxeles en la dimensión horizontal. En algunos NS-AIF y D-AIF convencionales, la simetría también existe diagonalmente entre las posiciones de subpíxeles en diferentes dimensiones. Por ejemplo, existe simetría de subpíxeles entre la posición de subpíxel "a" en la dimensión vertical y la posición de subpíxel "d" en la dirección horizontal en algunos NS-AIF y D-AIF convencionales. Con la simetría de subpíxeles en dimensiones horizontales, verticales y diagonales, se pueden utilizar hasta cinco conjuntos únicos de coeficientes de filtro de interpolación para una precisión de un cuarto de píxel.

Por otro lado, para algunos tipos de S-AIF, existe una simetría de subpíxeles entre las posiciones de subpíxeles en la dirección (o dimensión) vertical, pero no las posiciones de subpíxeles en la dimensión horizontal. En otras palabras, la simetría de subpíxeles no siempre existe para posiciones de subpíxeles en la misma dimensión. La simetría de subpíxeles tampoco existe en las dimensiones diagonales. La simetría de algunos esquemas S-AIF, por lo tanto, requiere más conjuntos de coeficientes de interpolación. En particular, para la precisión de un cuarto de píxel, algunos esquemas S-AIF requieren once conjuntos únicos de coeficientes de interpolación.

El esquema de simetría de subpíxeles descrito en esta divulgación, e ilustrado en la FIG. 7, puede dar como resultado una predicción más precisa que la simetría de subpíxeles descritas anteriormente para algunos tipos de NS-AIF y D-AIF. En particular, el esquema de simetría de subpíxeles de la FIG. 7 impone una simetría de subpíxeles en una dimensión (por ejemplo, en la dirección horizontal o vertical), pero no impone simetría de subpíxeles simultáneamente en ambas dimensiones (por ejemplo, en diagonal).

Al no imponer la simetría de subpíxeles para las posiciones de subpíxeles en diagonal, la débil correlación entre los coeficientes de filtro de interpolación y las posiciones de subpíxeles en diferentes dimensiones no está integrada en

la interpolación. Aunque pueden ser necesarios más conjuntos de coeficientes de interpolación (por ejemplo, ocho en lugar de cinco), los datos de predicción interpolados resultantes pueden ser más precisos. La simetría de coeficientes diagonales también se puede evitar o eliminar, de la misma manera.

- 5 La unidad de predicción 32 también puede imponer la simetría de coeficientes como se ha descrito anteriormente. En particular, se impone la simetría de coeficientes para los coeficientes de filtro en una dimensión (por ejemplo, la dirección horizontal o vertical), pero no para ambas dimensiones en diagonal. Por ejemplo, los coeficientes de filtro para la posición de subpíxel "e" no están obligados a ser diagonalmente simétricos como en el esquema NS-AIF descrito anteriormente y representado en la ecuación (11). La simetría de coeficientes se resume a continuación en las ecuaciones (17)-(21).

$$h_0^b = h_5^b, h_1^b = h_4^b, h_2^b = h_3^b \quad (17)$$

$$h_0^h = h_5^h, h_1^h = h_4^h, h_2^h = h_3^h \quad (18)$$

$$h_{i,j}^f = h_{i,3-j}^f, \text{ for } i, j = 0 \dots 3 \quad (19)$$

$$h_{i,j}^i = h_{3-i,j}^i, \text{ for } i, j = 0 \dots 3 \quad (20)$$

$$h_{i,j}^j = h_{i,3-j}^j = h_{3-i,j}^j = h_{3-i,3-j}^j, \text{ for } i, j = 0 \dots 3 \quad (21)$$

- 15 Ha de apreciarse que para las posiciones de subpíxeles "f", "i" y "j", algunos coeficientes de filtro, es decir,

$h_{0,0}^{SP} = h_{0,3}^{SP} = h_{3,0}^{SP} = h_{3,3}^{SP}$ son equivalentes a cero en casos en los que puede usarse el filtro de 12 posiciones descrito en detalle con respecto a la FIG. 4. Como tal, la simetría de subpíxeles y de coeficientes descrita en esta divulgación se puede usar junto con o separadamente de las técnicas de soporte de filtro. Cuando se utiliza junto con el soporte de filtro de 12 posiciones descrito en la FIG. 4, hay 6(a) + 3(b) + 6(d) + 3(h) + 12(e) + 6(f) + 6(i) + 3(j) = 45 coeficientes únicos para estas posiciones de subpíxeles que deben ser enviados al decodificador, es decir, 6 coeficientes para el grupo que incluye la posición de subpíxel "a", 3 coeficientes para el grupo que incluye la posición de subpíxel "b", 6 coeficientes para el grupo que incluye la posición de subpíxel "d", 3 coeficientes para el grupo que incluye la posición de subpíxel "h", 12 coeficientes para el grupo que incluye la posición de subpíxel "e", 6 coeficientes para el grupo que incluye la posición de subpíxel "f", 6 coeficientes para el grupo que incluye la posición de subpíxel "i", y 3 coeficientes para el grupo que incluye la posición de subpíxel "j".

La FIG. 17 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza un soporte de filtro de doce píxeles coherente con esta divulgación. La técnica de la FIG. 17 se describirá desde la perspectiva del codificador de vídeo 50 de la FIG. 2, aunque otros dispositivos pueden realizar técnicas similares. Como se muestra en la FIG. 17, la unidad MC 37 de la unidad de predicción 32 obtiene un bloque de píxeles de la memoria 34 que incluye valores de píxeles enteros correspondientes a posiciones de píxeles enteros (171). El filtro o filtros 39 calculan valores de subpíxeles basándose en un soporte de filtro que comprende doce o más posiciones que rodean las posiciones de subpíxeles (172). Como se explica con mayor detalle anteriormente, el soporte de filtro de doce posiciones puede utilizarse para generar nueve de las quince posibles interpolaciones de subpíxeles, mientras que el filtrado lineal horizontal y lineal vertical puede utilizarse para generar seis de las quince posibles interpolaciones de subpíxeles.

La unidad MC 37 genera entonces un bloque de predicción basado en los valores de subpíxeles calculados (173). En particular, la unidad MC 37 puede generar y emitir un bloque de predicción interpolado que comprende valores de subpíxeles interpolados. El sumador 48 puede entonces codificar un bloque de vídeo actual basado en el bloque de predicción interpolado (174), por ejemplo, restando el bloque de predicción interpolado del bloque de vídeo que se codifica para generar un bloque residual. El bloque residual puede entonces transformarse y cuantificarse por la unidad de transformada 38 y la unidad de cuantificación 40, respectivamente. Después de la codificación por entropía por la unidad de codificación por entropía 46, el codificador de vídeo 50 puede emitir una información de flujo de bits de vídeo codificado y de filtro (175). La información de filtro, como se describe en el presente documento, puede comprender ocho conjuntos de coeficientes utilizados para generar los quince conjuntos de coeficientes para quince posiciones de subpel. La información del filtro puede emitirse una vez por unidad codificada, o posiblemente varias veces por unidad codificada si diferentes áreas de una unidad codificada usan diferentes tipos de interpolación de subpel.

La FIG. 18 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza un soporte de filtro de doce píxeles coherente con esta divulgación. Por lo tanto, el proceso de la FIG. 18 puede

considerarse el proceso de decodificación recíproco al proceso de codificación de la FIG. 17. La FIG. 18 se describirá desde la perspectiva del decodificador de vídeo 60 de la FIG. 16, aunque otros dispositivos pueden realizar técnicas similares. Como se muestra en la FIG. 18, el decodificador de vídeo 60 recibe una información de bloques de vídeo codificados y de filtro (181). La unidad de decodificación por entropía 52 puede decodificar por entropía esta información recibida. La unidad de predicción 55 realiza una compensación de movimiento de interpolación de acuerdo con las técnicas de esta divulgación. En particular, la unidad de predicción 55 obtiene un bloque de píxeles de la memoria 62 que incluye valores de píxeles enteros correspondientes a posiciones de píxeles enteros (182). La unidad de predicción 55 puede usar vectores de movimiento recibidos para determinar cómo realizar la interpolación. En base a los vectores de movimiento, la unidad de predicción 55 puede calcular subpíxeles basándose en un soporte de filtro que comprende doce o más posiciones que rodean las posiciones de subpíxel (183). De esta manera, la unidad de predicción 55 utiliza la interpolación para generar el bloque de predicción (184). El bloque de predicción puede utilizarse para decodificar un bloque de vídeo residual recurriendo al sumador 64 para añadir el bloque de predicción al bloque residual (185). Los diversos aspectos de interpolación descritos en el presente documento, incluyendo el uso de un soporte de filtro de doce posiciones que rodea las posiciones de subpíxeles en una forma radial, pueden mejorar la codificación de vídeo proporcionando datos de interpolación de mejor calidad que las técnicas convencionales.

La FIG. 19 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza simetría de coeficientes y simetría de píxeles coherente con esta divulgación. La técnica de la FIG. 19 se describirá desde la perspectiva del codificador de vídeo 50 de la FIG. 2, aunque otros dispositivos pueden realizar técnicas similares. Como se muestra en la FIG. 19, la unidad de predicción 32 del codificador de vídeo 50 define ocho conjuntos de coeficientes de filtro para quince posiciones de subpíxel basadas en simetría de coeficientes y simetría de píxeles, como se describe en el presente documento (201). En particular, la unidad de predicción 32 puede utilizar simetría de píxeles para reducir los conjuntos de coeficientes de quince a ocho, y puede reducir aún más el número de coeficientes para algunos o los ocho conjuntos basándose en la simetría de coeficientes entre los coeficientes de conjuntos dados. La unidad de predicción 32 puede entonces codificar datos de vídeo usando filtros de interpolación (tales como el filtro o filtros 39) y los ocho conjuntos de coeficientes de filtro (202). El codificador de vídeo 50 puede emitir datos de vídeo codificados y los ocho conjuntos de coeficientes de filtro (203). Los ocho conjuntos de coeficientes, que están codificados y se emiten en el flujo de bits, se pueden enviar con cada unidad codificada (por ejemplo, cada trama o fragmento) para que un dispositivo de decodificación pueda ser informado de los coeficientes del filtro que se utilizarán para la interpolación. Como alternativa, se pueden codificar y enviar diferentes grupos de ocho conjuntos de coeficientes con cada unidad codificada para permitir diferentes tipos de interpolaciones en diferentes lugares en una unidad codificada.

La FIG. 20 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza simetría de coeficientes y simetría de píxeles coherente con esta divulgación. En este caso, la unidad de predicción 55 del decodificador de vídeo 60 recibe ocho conjuntos de coeficientes de filtro (191), y genera los quince conjuntos de coeficientes de filtro basándose en los ocho conjuntos, simetría de coeficientes y simetría de píxeles. La unidad de predicción 60 puede entonces programar sus filtros de interpolación y aplicar dichos filtros de interpolación para generar adecuadamente datos predictivos interpolados (193). El decodificador de vídeo 60 puede entonces decodificar bloques de vídeo basándose en los datos predictivos (194), por ejemplo, recurriendo al sumador 64 para añadir datos predictivos correctos interpolados por la unidad de predicción 55 a un bloque residual a decodificar.

La FIG. 21 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un codificador de vídeo que utiliza el filtrado de ubicaciones de píxeles enteros para generar valores de píxeles enteros ajustados consistentes con esta divulgación. Esto se ha analizado anteriormente como filtrado de tipo interpolación en la medida en que es similar a las interpolaciones, pero no genera valores subenteros. Más bien, este proceso genera nuevos valores enteros que se filtran en función del valor entero original y otros valores enteros que rodean el valor entero original.

La FIG. 21 se describirá desde la perspectiva del codificador de vídeo 50 de la FIG. 2, aunque otros dispositivos pueden realizar técnicas similares. Como se muestra en la FIG. 21, la unidad MC 37 de la unidad de predicción 32 obtiene un bloque de píxeles de la memoria 34 que incluye valores de píxeles enteros correspondientes a ubicaciones de píxeles enteros (211). El filtro o filtros 39 de la unidad de predicción 32 filtran los valores de píxeles enteros basándose en otros valores de píxeles enteros del bloque de píxeles para generar valores de píxeles enteros ajustados (212). La unidad de predicción 32 genera un bloque de predicción basado en los valores de píxeles enteros ajustados (213), y el codificador de vídeo 50 codifica un bloque de vídeo basado en el bloque de predicción (214), por ejemplo, recurriendo al sumador 48 para restar el bloque de predicción del bloque de vídeo que se está codificando para generar un bloque residual. Después de la transformación y cuantificación del bloque residual mediante la unidad de transformada 38 y la unidad de cuantificación 40, y después de la codificación por entropía por la unidad de codificación por entropía 46, el codificador de vídeo 50 envía la información de bloque de vídeo codificado y de filtro (215). Como con la interpolación subpel, el filtrado de tipo interpolación con respecto a las ubicaciones de píxeles enteros puede implicar la salida y comunicación de la información de filtro una vez por unidad codificada, o posiblemente varias veces por unidad codificada si diferentes áreas de una unidad codificada usan diferentes tipos de filtrado de tipo interpolación de enteros.

La FIG. 22 es un diagrama de flujo que ilustra el funcionamiento ejemplar de un decodificador de vídeo que utiliza el

filtrado de ubicaciones de píxeles enteros para generar valores de píxeles enteros ajustados consistentes con esta divulgación. La unidad de predicción 55 del decodificador de vídeo 60 recibe bloques de vídeo codificados y también recibe información de filtro (por ejemplo, una vez por unidad codificada) (221). La unidad de predicción 55 obtiene un bloque de píxeles de la memoria 62 que incluye valores de píxeles enteros correspondientes a posiciones de píxeles enteros (222). La unidad de predicción 55 recurre a un filtro (no mostrado) como parte de la compensación de movimiento para filtrar los valores de píxeles enteros basándose en otros valores de píxeles enteros del bloque de píxeles para generar valores de píxeles enteros ajustados (223). La unidad de predicción 55 genera un bloque de predicción basado en los valores de píxeles enteros ajustados (224). El decodificador de vídeo 60 puede entonces decodificar bloques de vídeo basándose en el bloque de predicción, por ejemplo, recurriendo al sumador 64 para añadir el bloque predictivo interpolado por la unidad de predicción 55 a un bloque de vídeo residual a decodificar (225).

La FIG. 23 es un diagrama de flujo que ilustra una técnica para la interpolación definida por distorsión de velocidad para codificación de vídeo basada en dos conjuntos de filtros candidatos. En este caso, la unidad MC 37 de la unidad de predicción 32 puede generar dos o más conjuntos diferentes de datos predictivos, un conjunto basado en el primer conjunto de filtros de interpolación candidatos y otro conjunto basado en el segundo conjunto de filtros de interpolación candidatos. La unidad MC 37 puede entonces utilizar un análisis basado en distorsión de velocidad para seleccionar qué filtro de interpolación producirá los mejores resultados desde la perspectiva de la velocidad y la distorsión. De esta manera, la unidad MC 37 no sólo considera qué conjunto de datos predictivos producirá los mejores resultados (es decir, la menor distorsión), sino que también tendrá en cuenta el hecho de que un conjunto de filtros candidatos puede requerir menos sobrecarga de bits para transportar sus coeficientes de filtro al decodificador.

Como se muestra en la FIG. 23, la unidad MC 37 de la unidad de predicción 32 genera los primeros datos predictivos interpolados para codificar datos de vídeo basándose en un primer filtro de interpolación (231). El primer filtro de interpolación, en un ejemplo, comprende un filtro de interpolación fijo que corresponde a un filtro definido por un estándar de codificación de vídeo particular, tal como un filtro que corresponde a un filtro de interpolación ITU-T H.264. Para mayor precisión, en algunos casos, el filtro de interpolación fijo puede corresponder a un filtro de interpolación ITU-T H.264 sin redondeo intermedio de valores de medio píxel que se utilizan para generar valores de cuarto de píxel. En este caso, el filtro de interpolación fijo puede generar valores de medio píxel y puede redondear hacia arriba dichos valores con fines de interpolación a la resolución de medio píxel. Sin embargo, en la medida en que tales valores de medio píxel también se usarán para la interpolación a la resolución de cuarto de píxel, la unidad MC 37 puede almacenar versiones no redondeadas de los valores de medio píxel y utilizar las versiones no redondeadas de los valores de medio píxel para cualquier interpolación a la resolución de cuarto de píxel. La Solicitud de Patente de Estados Unidos pendiente junto con la presente y comúnmente asignada n.º de serie _____, presentada por Karczewicz et al., y titulada "ADVANCED INTERPOLATION TECHNIQUES FOR MOTION COMPENSATION IN VIDEO CODING" con el número de registro 081399U1, que se presentó el mismo día que la presente solicitud, y que se incorpora en el presente documento por referencia, proporciona muchos detalles adicionales sobre el filtrado de interpolación sin redondeo intermedio de valores de medio píxel que se usan para generar valores de cuarto de píxel.

A continuación, la unidad MC 37 de la unidad de predicción 32 genera los segundos datos predictivos interpolados para codificar vídeo de los datos de vídeo basándose en un segundo filtro de interpolación (232). En un ejemplo, el segundo filtro de interpolación puede comprender un filtro de interpolación adaptativo. En este caso, en consonancia con la interpolación adaptativa, la unidad MC 37 puede definir los coeficientes de filtro que se van a utilizar. En particular, la unidad MC 37 puede ejecutar un proceso de interpolación adaptativa en el que la unidad MC 37 comienza con coeficientes de filtro preestablecidos, genera datos predictivos preliminares y luego ajusta los coeficientes de filtro en un proceso adaptativo para que dichos coeficientes de filtro definan datos predictivos más deseables.

Una vez que la unidad MC 37 de la unidad de predicción 32 ha generado tanto los primeros como segundos datos predictivos interpolados, la unidad MC 37 puede seleccionar entre los primeros datos predictivos interpolados y los segundos datos predictivos interpolados en base a un análisis de distorsión de velocidad (233). De esta manera, la unidad MC 37 no sólo considera qué conjunto de datos predictivos producirá los mejores resultados (es decir, la menor distorsión), sino que la unidad MC 37 también tiene en cuenta las diferentes cantidades de datos (es decir, las diferentes velocidades de bits) que se requerirán para el primer filtro de interpolación con respecto al segundo filtro de interpolación. En particular, si se utiliza el filtro de interpolación fijo (por ejemplo, como primer filtro de interpolación), el codificador de vídeo 50 no necesitará codificar coeficientes de filtro en el flujo de bits, mientras que si se utiliza el filtro de interpolación adaptativo (por ejemplo, como el segundo filtro de interpolación), el codificador de vídeo 50 necesitará codificar coeficientes de filtro. En consecuencia, el análisis de distorsión de velocidad puede determinar qué conjunto de datos predictivos producirá los mejores resultados (es decir, el menor coste de distorsión de velocidad), teniendo en cuenta el hecho de que el uso del filtro de interpolación fijo no requiere bits adicionales para transmitir coeficientes de filtro al decodificador.

Más específicamente, la selección entre los primeros datos predictivos interpolados y los segundos datos predictivos interpolados basándose en un análisis de distorsión de velocidad (233) puede comprender el cálculo de un primer

coste de distorsión de velocidad asociado con los datos de vídeo si los datos de vídeo se codifican a través de los primeros datos predictivos interpolados, calcular un segundo coste de distorsión de velocidad asociado con los datos de vídeo si los datos de vídeo se codifican a través de los segundos datos predictivos interpolados, y seleccionar entre los primeros datos predictivos interpolados y los segundos datos predictivos interpolados en base a primer y segundo coste de distorsión de velocidad.

El coste de distorsión de velocidad asociado con los datos de vídeo si los datos de vídeo están codificados a través de los primeros datos predictivos interpolados puede comprender una primera métrica de diferencia, por ejemplo, error cuadrático medio (MSE) de valores de píxeles o suma de diferencia absoluta (SAD) de valores de píxeles o suma de diferencia cuadrática (SSD) de valores de píxeles indicativos de diferencias entre los datos de vídeo y los primeros datos predictivos interpolados, más un primer valor que cuantifica el coste asociado con la codificación de coeficientes de filtro. En este caso, si el primer filtro de interpolación es un filtro de interpolación fijo, el primer valor que cuantifica el coste puede definirse como cero. De forma similar, el segundo coste de distorsión de velocidad puede comprender una segunda métrica de diferencia indicativa de diferencias entre los datos de vídeo y los segundos datos predictivos interpolados (MSE, SAD o SSD), más un segundo valor que cuantifica el coste asociado con la codificación de coeficientes de filtro. En el caso en el que el segundo filtro de interpolación es un filtro de interpolación adaptativo, el segundo valor que cuantifica el coste asociado con la codificación de coeficientes de filtro puede comprender el número de bits (r) necesario para codificar coeficientes de filtro de interpolación adaptativos, o posiblemente este número (r) multiplicado por un multiplicador lagrangiano (λ).

Después de seleccionar entre los primeros datos predictivos interpolados y los segundos datos predictivos interpolados en base a un análisis de distorsión de velocidad (233), la unidad MC 37 de la unidad de predicción 32 codifica los datos de vídeo basados en la selección (234) y codifica la sintaxis para indicar la selección (235). La sintaxis puede comprender un indicador de un bit o un indicador de múltiples bits que define si el decodificador debe usar el primer filtro de interpolación o el segundo filtro de interpolación. El proceso de la FIG. 23 puede repetirse para cada una de una pluralidad de ubicaciones de subpíxeles para cada unidad codificada de una secuencia de vídeo para indicar si el primer filtro de interpolación o el segundo filtro de interpolación deberían ser utilizados por un decodificador para cada una de la pluralidad de ubicaciones de subpíxeles. Las ubicaciones de subpíxeles pueden comprender quince posibles ubicaciones de subpíxeles coherentes con la interpolación a la resolución de cuarto de píxel, o pueden comprender un número diferente de ubicaciones de subpíxeles. El proceso de la FIG. 23 también puede repetirse para la posición de píxeles enteros para cada unidad codificada de una secuencia de vídeo para indicar si el primer filtro de interpolación o el segundo filtro de interpolación deben ser utilizados por un decodificador para la ubicación de píxeles enteros.

Una vez que el codificador de vídeo 50 (FIG. 2) ha codificado los datos de vídeo basados en la selección (234), y la sintaxis codificada para indicar la selección (235), el módem 23 y el transmisor 24 (FIG. 1) pueden modular y transmitir los datos de vídeo codificados y la sintaxis al dispositivo de destino 17. En el caso en que el primer filtro de interpolación es fijo y el segundo filtro de interpolación es adaptativo, el transmisor 24 puede transmitir coeficientes de filtro cuando la sintaxis indica que los segundos datos predictivos interpolados se utilizaron para generar los datos de vídeo codificados, pero no pueden transmitir coeficientes de filtro cuando la sintaxis indica que los primeros datos predictivos interpolados se utilizaron para generar los datos de vídeo codificados. De esta manera, cuando el primer filtro de interpolación es fijo y el segundo filtro de interpolación es adaptativo, los coeficientes de filtro sólo se envían si la sintaxis indica que se utilizó el filtrado de interpolación adaptativa, y la decisión de usar o no el filtrado adaptativo de interpolación considera no sólo la calidad de vídeo predictiva, sino también la velocidad de bits, que se ve afectada por la presencia de coeficientes de filtro en el flujo de bits transmitido. Sin embargo, en otros ejemplos, el primer y segundo filtros de interpolación pueden ser ambos fijos, o ambos pueden ser adaptativos.

Las técnicas de esta divulgación pueden aplicarse en muchos escenarios, incluyendo escenarios cuando hay más de dos conjuntos de filtros seleccionados por el codificador. En otras palabras, también se pueden aplicar filtros de interpolación adicionales para generar datos predictivos interpolados adicionales, que también se pueden considerar en el análisis de distorsión de velocidad. En otras palabras, el método no se limita a generar sólo primeros y segundos datos predictivos interpolados basados en dos filtros de interpolación, sino que podría aplicarse para generar cualquier pluralidad de datos predictivos interpolados basados en cualquier número de filtros de interpolación. Es importante destacar que se usan datos predictivos interpolados por análisis de distorsión de velocidad para identificar qué filtro seleccionar. En un ejemplo, un método ejecutado por el codificador de vídeo 50 puede comprender generar una pluralidad de versiones diferentes de datos predictivos para codificar datos de vídeo basados en una pluralidad de diferentes filtros de interpolación, seleccionar entre la pluralidad de versiones diferentes de datos predictivos basados en un análisis de distorsión de velocidad, codificar los datos de vídeo basados en la selección, y codificar la sintaxis para indicar la selección.

Además de seleccionar el filtro de interpolación basado en un análisis de distorsión de velocidad, y generar sintaxis para indicar la selección, la unidad MC 37 de la unidad de predicción 32 también puede llevar a cabo un análisis similar de distorsión de velocidad con respecto al filtrado de píxeles enteros, que se ha explicado en más detalle anteriormente. En particular, la unidad MC 37 de la unidad de predicción 32 puede generar dos o más conjuntos diferentes de datos predictivos para ubicaciones de píxeles enteros, por ejemplo, uno con filtrado de píxeles enteros y uno sin filtrado de píxeles enteros, y puede llevar a cabo un análisis de distorsión de velocidad con respecto a

estos dos conjuntos diferentes con el fin de determinar si se desea un filtrado de píxeles enteros. Por consiguiente, la unidad MC 37 de la unidad de predicción 32 puede generar otro elemento sintáctico basado en un análisis de distorsión de velocidad asociado con dos conjuntos diferentes de datos predictivos de enteros para indicar si se debe aplicar filtrado de píxeles enteros para la unidad codificada, en la que los datos predictivos de enteros no se filtran y se filtra un segundo conjunto de datos predictivos de enteros. De esta manera, la decisión de si llevar a cabo filtrado de píxeles enteros puede basarse, no sólo en la calidad de la codificación de vídeo, sino también en la posible sobrecarga de bits asociada con el filtrado de píxeles enteros, particularmente cuando el filtrado de píxeles enteros implica la codificación y el envío de coeficientes de filtro de píxeles enteros utilizados para realizar dicho filtrado de píxeles enteros. El filtrado de enteros también puede considerar N filtros de enteros (por ejemplo, donde N es cualquier número entero positivo y plural. De acuerdo con el uso de N filtros de enteros, el ejemplo anterior corresponderá al caso en el que N es dos y uno de los filtros no aplica ningún filtrado.

Siempre que los coeficientes de filtro de interpolación están realmente codificados y enviados desde el dispositivo de origen 12 al dispositivo de destino 16, esta divulgación también contempla técnicas para codificar tales coeficientes de filtro de interpolación. La codificación descrita para los coeficientes de filtro puede mejorar la compresión de datos. En particular, esta divulgación contempla técnicas de predicción para coeficientes de filtro, por ejemplo, con respecto a coeficientes de filtro fijo. Además, esta divulgación contempla técnicas de predicción para un segundo conjunto de coeficientes de filtro con relación al primer conjunto de coeficientes de filtro. De esta manera, se puede aprovechar la simetría imperfecta entre diferentes coeficientes de filtro para permitir la compresión de datos. Además de la utilización de dichas técnicas de predicción para coeficientes de filtro de interpolación, esta divulgación también proporciona técnicas útiles de cuantificación y codificación por entropía de coeficientes de filtro de interpolación en base a la codificación de prefijo y sufijo. El siguiente análisis proporciona más detalles sobre estos aspectos de esta divulgación.

La FIG. 24 es un diagrama de flujo que ilustra una técnica para codificar coeficientes de filtro usando codificación predictiva. En este caso, los coeficientes de filtro utilizados por el codificador de vídeo 50 pueden codificarse de forma predictiva con relación a los coeficientes de filtro de un filtro fijo, lo que puede mejorar aún más la compresión de datos cuando se envían coeficientes de filtro como parte de un flujo de bits codificado.

Como se muestra en la FIG. 24, la unidad MC 37 de la unidad de predicción 32 identifica un conjunto de coeficientes de filtro para la interpolación de datos predictivos en codificación de vídeo (241). Por ejemplo, la unidad MC 37 puede identificar el conjunto de coeficientes de filtro realizando un proceso de filtración de interpolación adaptativa. En este caso, la unidad MC 37 puede determinar el conjunto de coeficientes de filtro a través del proceso de filtrado de interpolación adaptativa y generar los datos predictivos basados en los coeficientes de filtro identificados en el proceso de filtrado por interpolación adaptativa. En el filtrado de interpolación adaptativa, tal como se explica con mayor detalle en el presente documento, la unidad 37 de MC puede realizar un enfoque de dos pasadas en el que se generan datos predictivos basados en un filtro fijo y luego los coeficientes de filtro se ajustan para que los datos predictivos se hagan más similares a los datos de vídeo que se codifican. Los coeficientes de filtro ajustados, entonces, definen los coeficientes de filtro que se utilizan y codifican en el flujo de bits.

Con el fin de codificar los coeficientes de filtro para que tales coeficientes de filtro se puedan transmitir como parte de un flujo de bits, la unidad de predicción y cuantificación de coeficientes 41 puede generar valores residuales asociados con el conjunto de coeficientes de filtro basados en la codificación predictiva del conjunto de coeficientes de filtro con respecto a coeficientes de filtro asociados con un filtro de interpolación fijo (242). En particular, la unidad de predicción y cuantificación de coeficientes 41 puede restar los coeficientes de filtro determinados en el proceso de filtrado de interpolación adaptativa a partir de coeficientes de filtro correspondientes asociados a un filtro fijo, tal como un filtro de interpolación ITU-T H.264, o un filtro de interpolación ITU-T H.264 sin redondeo intermedio de valores de medio píxel. Mediante la codificación y transmisión de valores residuales, en lugar de transmitir los coeficientes de filtro reales, se puede reducir la cantidad de datos comunicados en el flujo de bits. En este caso, el decodificador puede programarse para conocer la manera en la que están codificados los coeficientes de filtro.

El codificador de vídeo 50 puede recurrir a la unidad de predicción y cuantificación de coeficientes 41 tanto para predecir como cuantificar los coeficientes residuales, y la unidad de codificación por entropía 46 puede codificar por entropía los residuos cuantificados (243). El codificador de vídeo 50 puede entonces emitir los valores residuales como parte del flujo de bits codificado (244). La cuantificación de valores residuales asociados con los coeficientes de filtro puede implicar la cuantificación de los valores residuales, en los que al menos algunos de los valores residuales asociados con diferentes coeficientes de filtro se asignan a diferentes niveles de cuantificación. De esta manera, la unidad de predicción y cuantificación de coeficientes 41 puede asignar más cuantificación a coeficientes residuales más grandes, y puede asignar menos cuantificación a coeficientes residuales más finos para conseguir un equilibrio deseable de cuantificación y precisión. Cuando se utiliza más cuantificación, se eliminan más datos pero se puede conseguir más compresión. La unidad de codificación por entropía 46 puede codificar por entropía los valores residuales asignando un código de prefijo, asignando un código de sufijo y asignando un valor de signo para los valores residuales después de la cuantificación de los valores residuales. El codificador de vídeo 50 puede entonces emitir los valores residuales como parte del flujo de bits codificado (244).

La FIG. 27 es un gráfico conceptual que ilustra algunos coeficientes de filtro ilustrativos que pueden codificarse de

forma predictiva. En este caso, los coeficientes de filtro O_1 , O_2 , O_3 y O_4 pueden definir coeficientes de filtro asociados con un filtro de interpolación fijo. Los coeficientes de filtro X_1 , X_2 , X_3 y X_4 pueden definir los coeficientes de filtro deseados, que pueden ser bastante similares a los del filtro de interpolación fijo. Por consiguiente, los coeficientes de filtro X_1 , X_2 , X_3 y X_4 pueden predecirse basándose en los coeficientes de filtro O_1 , O_2 , O_3 y O_4 respectivamente.

5 En particular, se puede formar un primer residuo como la diferencia entre X_1 y O_1 . De forma similar, puede formarse un segundo residuo como la diferencia entre X_2 y O_2 , se puede formar un tercer residuo como la diferencia entre X_3 y O_3 , y se puede formar un cuarto residuo como la diferencia entre X_4 y O_4 . Los residuos pueden comprender menos datos que los coeficientes de filtro originales, promoviendo así la compresión de datos.

10 En algunos casos, el conjunto de coeficientes de filtro comprende un primer conjunto de coeficientes de filtro que definen solamente una parte de un conjunto completo de coeficientes de filtro asociados con un filtro de interpolación adaptativo, en el que un segundo conjunto de coeficientes de filtro puede determinarse mediante un decodificador basado en el primer conjunto de coeficientes de filtro y simetría de coeficientes. Por ejemplo, los coeficientes de filtro X_1 y X_2 pueden codificarse de forma predictiva basándose en O_1 y O_2 respectivamente. En este caso, sin embargo, X_3 y X_4 pueden ser simétricos con X_1 y X_2 y el decodificador puede programarse para saber que dicha simetría está impuesta. Por consiguiente, a través del uso de simetría, los coeficientes X_3 y X_4 en este sencillo ejemplo, pueden eliminarse del flujo de bits, y pueden calcularse en el decodificador basándose en la simetría de coeficiente conocida una vez que los coeficientes X_1 y X_2 se decodifican predictivamente.

20 La FIG. 25 es otro diagrama de flujo que ilustra una técnica para codificar coeficientes de filtro usando codificación predictiva. En este caso, sin embargo, se utilizan dos tipos diferentes de predicción. Como se muestra en la FIG. 25, la unidad MC 37 de la unidad de predicción 32 identifica un conjunto de coeficientes de filtro para la interpolación de datos predictivos en codificación de vídeo (251). Como con el ejemplo de la FIG. 24, en la FIG. la unidad MC 37 puede identificar el conjunto de coeficientes de filtro realizando un proceso de filtración de interpolación adaptativa.

25 En este caso, la unidad MC 37 puede determinar el conjunto de coeficientes de filtro a través del proceso de filtrado de interpolación adaptativa y generar los datos predictivos basados en los coeficientes de filtro identificados en el proceso de filtrado por interpolación adaptativa. En el filtrado de interpolación adaptativa, la unidad 37 de MC puede realizar un enfoque de dos pasadas en el que se generan datos predictivos basados en un filtro fijo y luego los coeficientes de filtro se ajustan para que los datos predictivos se hagan más similares a los datos de vídeo que se codifican. Los coeficientes de filtro ajustados, entonces, definen los coeficientes de filtro que se utilizan y codifican en el flujo de bits.

Con el fin de codificar los coeficientes de filtro para que tales coeficientes de filtro se puedan transmitir como parte de un flujo de bits, la unidad de predicción y cuantificación de coeficientes 41 puede generar un primer conjunto de valores residuales asociado con un primer conjunto de los coeficientes de filtro basados en la codificación predictiva con respecto a coeficientes de filtro asociados con un filtro de interpolación fijo (252). En particular, la unidad de predicción y cuantificación de coeficientes 41 puede restar el primer conjunto de coeficientes de filtro de coeficientes de filtro correspondientes asociados a un filtro fijo, tal como un filtro de interpolación ITU-T H.264, o un filtro de interpolación ITU-T H.264 sin redondeo intermedio de valores de medio píxel. A continuación, la unidad de predicción y cuantificación de coeficientes 41 puede generar un segundo conjunto de valores residuales asociados con un segundo conjunto de coeficientes de filtro basados en la codificación predictiva con respecto al primer conjunto de coeficientes de filtro (253). En particular, la unidad de predicción y cuantificación de coeficientes 41 puede restar el segundo conjunto de coeficientes de filtro de los valores reflejados o rotados del primer conjunto de coeficientes. Por lo tanto, el primer conjunto de coeficientes se codifica de forma predictiva basándose en los coeficientes de un filtro fijo y el segundo conjunto de coeficientes se codifica de forma predictiva en base al primer conjunto de coeficientes. Mediante la generación de valores residuales, en lugar de usar los coeficientes de filtro reales, se puede reducir la cantidad de datos comunicados en el flujo de bits. Además, utilizando el filtro fijo para predecir el primer conjunto de coeficientes y después usando el primer conjunto de coeficientes para predecir el segundo conjunto de coeficientes, se puede conseguir una compresión de datos adicional con respecto a la predicción que se basa únicamente en el filtro fijo.

Como con el ejemplo de la FIG. 24, en la FIG. 25, el codificador de vídeo 50 puede recurrir a la unidad de predicción y cuantificación de coeficientes 41 tanto para codificar predictivamente como cuantificar los coeficientes residuales, y la unidad de codificación por entropía 46 puede codificar por entropía los residuos cuantificados (254). El codificador de vídeo 50 puede entonces emitir los valores residuales como parte del flujo de bits codificado (254). De nuevo, la cuantificación de valores residuales asociados con los coeficientes de filtro puede implicar la cuantificación de los valores residuales, en los que al menos algunos de los valores residuales asociados con diferentes coeficientes de filtro se asignan a diferentes niveles de cuantificación. De esta manera, la unidad de predicción y cuantificación de coeficientes 41 puede asignar más cuantificación a coeficientes residuales más grandes, y puede asignar menos cuantificación a coeficientes residuales más finos para conseguir un equilibrio deseable de cuantificación y precisión. La unidad de codificación por entropía 46 puede codificar por entropía los valores residuales asignando un código de prefijo, asignando un código de sufijo y asignando un valor de signo para los valores residuales después de la cuantificación de los valores residuales. El codificador de vídeo 50 puede entonces emitir los valores residuales como parte del flujo de bits codificado (255).

65 La FIG. 28 es un gráfico conceptual que ilustra algunos coeficientes de filtro ilustrativos que pueden codificarse de

forma predictiva coherente con el proceso de la FIG. 25. En este caso, los coeficientes de filtro O_1 y O_2 pueden definir coeficientes de filtro asociados con un filtro de interpolación fijo. Los coeficientes de filtro X_1 y X_2 pueden definir un primer conjunto de los coeficientes de filtro deseados, que pueden ser bastante similares a los del filtro de interpolación fijo. Por consiguiente, los coeficientes de filtro X_1 y X_2 pueden predecirse basándose en los coeficientes de filtro O_1 y O_2 respectivamente. En particular, puede formarse un primer residuo como la diferencia entre X_1 y O_1 , y un segundo residuo puede formarse como la diferencia entre X_2 y O_2 . Los residuos pueden comprender menos datos que los coeficientes de filtro originales, promoviendo así la compresión de datos. Los residuos se cuantifican entonces por la unidad de predicción y cuantificación de coeficientes 41 y se codifican por entropía por la unidad de codificación por entropía 46. \hat{X}_1 y \hat{X}_2 pueden referirse a coeficientes de filtro modificados que se generan sumando los residuos descuantificados a los coeficientes de filtro de predicción O_1 y O_2 .

A continuación, se puede codificar predictivamente un segundo conjunto de coeficientes de filtro Z_1 y Z_2 basándose en el primer conjunto de coeficientes X_1 y X_2 , por ejemplo, específicamente a partir de los coeficientes \hat{X}_1 y \hat{X}_2 que se definen basándose en los coeficientes X_1 y X_2 . En particular, puede formarse un tercer residuo como la diferencia entre Z_1 y \hat{X}_1 , y puede formarse un cuarto residuo como la diferencia entre Z_2 y \hat{X}_2 . \hat{X}_1 y \hat{X}_2 pueden ser más similares a Z_1 y Z_2 que O_1 y O_2 y, por lo tanto, utilizando \hat{X}_1 y \hat{X}_2 para codificar predictivamente Z_1 y Z_2 se puede promover una compresión de datos adicional.

La FIG. 26 es un diagrama de flujo que ilustra una técnica para decodificar coeficientes de filtro usando codificación predictiva. La FIG. 26 se describirá desde la perspectiva del decodificador de vídeo 60 de la FIG. 16. Como se muestra, el decodificador de vídeo 60 recibe valores residuales asociados con un conjunto de coeficientes de filtro (261). El decodificador de vídeo 60 puede decodificar por entropía los valores residuales a través de la unidad de decodificación por entropía 52 y puede recurrir a la unidad de predicción de coeficientes y cuantificación inversa 53 para descuantificar los valores residuales (262), los cuales se envían a la unidad de predicción 55. La unidad de predicción 56 genera el conjunto de coeficientes de filtro utilizando la decodificación predictiva de los valores residuales (263).

En particular, la unidad de predicción 56 puede generar todo el conjunto de coeficientes de filtro basándose en los valores residuales y coeficientes de filtro asociados con un filtro de interpolación fijo, por ejemplo, como se ilustra conceptualmente en la FIG. 27 y se ha abordado anteriormente en el contexto de la codificación. En algunos casos, se puede generar un primer conjunto de coeficientes de filtro basándose en los valores residuales y los coeficientes de filtro asociados con un filtro de interpolación fijo, y se puede generar un segundo conjunto de coeficientes de filtro basado en la simetría. En otros casos, se puede generar un primer conjunto de coeficientes de filtro basándose en los valores residuales y coeficientes de filtro asociados a un filtro de interpolación fijo, y se puede generar un segundo conjunto de coeficientes de filtro en base a los valores residuales adicionales y el primer conjunto de coeficientes de filtro, por ejemplo, como se ilustra conceptualmente en la FIG. 28 y se ha abordado anteriormente en el contexto de la codificación. En cualquier caso, la unidad de predicción 56 del decodificador de vídeo 60 aplica el conjunto de coeficientes de filtro para interpolar los datos predictivos utilizados para la decodificación predictiva de los bloques de vídeo (264). En particular, la unidad de predicción 56 filtra los datos para generar datos predictivos interpolados utilizando los coeficientes de filtro decodificados de forma predictiva, de manera que los bloques de vídeo se pueden decodificar basándose en dichos datos predictivos interpolados.

De nuevo, el conjunto de coeficientes de filtro decodificados predictivamente puede comprender un primer conjunto de coeficientes de filtro que definen sólo una porción de un conjunto completo de coeficientes de filtro asociados con un filtro de interpolación adaptativo. En este caso, la unidad de predicción de coeficientes y de cuantificación inversa 53 del decodificador de vídeo 60 puede generar un segundo conjunto de coeficientes de filtro basado en el primer conjunto de coeficientes de filtro y simetría de coeficientes y aplicar los conjuntos primero y segundo de coeficientes de filtro para interpolar los datos predictivos.

En otro caso, el conjunto de coeficientes de filtro decodificados predictivamente puede comprender un primer conjunto de coeficientes de filtro que definen sólo una porción de un conjunto completo de coeficientes de filtro asociados con un filtro de interpolación adaptativo. En este caso, el decodificador de vídeo 60 puede recibir valores residuales adicionales asociados con todo el conjunto de coeficientes de filtro. La unidad de predicción de coeficientes y de cuantificación inversa 53 puede generar un segundo conjunto de coeficientes de filtro usando decodificación predictiva basada en valores residuales adicionales y el primer conjunto de coeficientes de filtro, y la unidad de predicción 55 puede aplicar el primer y segundo conjuntos de coeficientes de filtro para interpolar los datos predictivos.

En algunos casos, pueden usarse filtros de interpolación fijos basados en filtros H.264/AVC para predecir los filtros 1-D (que pueden incluir filtros para las posiciones de subpíxel a, b, d, h mostradas en las FIGS. 8, 9, 10 y 11). Para los filtros 2-D, que incluyen filtros para las posiciones de subpíxel e, f, i, y j mostradas en las FIGS. 12, 13, 14 y 15, puede utilizarse uno de los siguientes esquemas de predicción:

1. Establecer predicción a cero (sin predicción), $p(h_{i,j}^{SP}) = 0$, $SP \in \{e, f, i, j\}$

65

2. Utilizar una predicción de filtro fijo, tal como el filtro promedio reunido sobre un conjunto de entrenamiento, es decir, $p(h_{i,j}^{SP}) = \bar{h}_{i,j}^{SP}$, $SP \in \{e, f, i, j\}$, donde $\bar{h}_{i,j}^{SP}$ es el coeficiente de filtro (i, j)-ésimo en el filtro promedio para la posición de subpíxel SP.

5 3. Aprovechar la posible simetría de los coeficientes y utilizar los coeficientes ya codificados para predecir los coeficientes restantes.

Para los filtros 1-D, también puede aplicarse cualquiera de estos tres métodos de predicción.

10 Para la predicción, la FIG. 29 proporciona un ejemplo de una matriz de coeficientes de filtro de píxeles enteros para los que pueden usarse técnicas de predicción para la codificación. En este ejemplo, puede suponerse que no se impone ninguna simetría en el filtro de píxeles enteros. Los coeficientes (h0,0), (h0,1), (h1,0), (h1,1), (h2,0), (h2,1) (h0,2), (h1,2) y (h2,2) se pueden cuantificar y codificar primero. A continuación, se pueden utilizar los coeficientes de la parte superior izquierda (h0,0), (h0,1), (h1,0), (h1,1) y (h2,0) ya codificados para predecir los coeficientes de la parte superior derecha (h0,3), (h1,3), (h2,3), (h0,4), (h1,4) y (h2,4). A continuación, una vez que la mitad superior de los coeficientes del filtro (h0,0), (h0,1), (h0,2), (h0,3) (h0,4), (h1,0), (h1,1), (h1,2), (h1,3) y (h1,4) se cuantifican y codifican, pueden utilizarse además para predecir la mitad inferior de los coeficientes de filtro (h3,0), (h3,1), (h3,2), (h3,3), (h3,4), (h4,0), (h4,1), (h4,2), (h4,3) y (h4,4). La predicción de otros coeficientes de filtro puede hacerse de una manera similar. Por ejemplo, para el filtro de posición "e" de subpíxel (véase la FIG. 12), que puede tener cierta simetría diagonal, los coeficientes de la parte superior derecha pueden cuantificarse y codificarse primero, y luego utilizarse para predecir los coeficientes de la parte inferior izquierda.

25 En cualquier caso, después de la predicción de los coeficientes (por ejemplo, mediante la unidad de predicción y cuantificación de coeficientes 41 de la unidad de predicción 32), los errores de predicción se cuantifican (por ejemplo, mediante la unidad de predicción y cuantificación de coeficientes 41). Como se ha descrito anteriormente, puede usarse la denominada "cuantificación desigual". En este caso, la precisión de cuantificación aplicada por la unidad de predicción y cuantificación de coeficientes 41 puede depender de la ubicación del coeficiente. Se ha encontrado que para coeficientes de menor magnitud (que son típicamente coeficientes más alejados del centro del filtro), puede ser deseable una mejor precisión. Por el contrario, los coeficientes de mayor magnitud (que son típicamente coeficientes más cercanos al centro del filtro), es más deseable una menor precisión.

35 Las siguientes matrices, Q^{1D} , Q^{2D} , Q^{FP} , pueden usarse por la unidad de predicción y cuantificación de coeficientes 41 para especificar la precisión de cuantificación para los coeficientes en los filtros 1D, los filtros 2D y el filtro de píxeles enteros, respectivamente. Obsérvese que los números de bits dados en las matrices pueden incluir 1 bit para codificar el signo de los respectivos coeficientes.

$$Q^{1D} = [12 \ 11 \ 9 \ 9 \ 11 \ 12]$$

$$Q^{2D} = \begin{bmatrix} 0 & 10 & 10 & 0 \\ 10 & 9 & 9 & 10 \\ 10 & 9 & 9 & 10 \\ 0 & 10 & 10 & 0 \end{bmatrix}$$

$$Q^{FP} = \begin{bmatrix} 11 & 11 & 11 & 11 & 11 \\ 11 & 10 & 10 & 10 & 11 \\ 11 & 10 & 9 & 10 & 11 \\ 11 & 10 & 10 & 10 & 11 \\ 11 & 11 & 11 & 11 & 11 \end{bmatrix}$$

40 La unidad de predicción y cuantificación de coeficientes 41 puede codificar los errores de predicción de coeficientes cuantificados, es decir, los residuos de coeficientes, basándose en un esquema de codificación de prefijo simple. En primer lugar, el rango dinámico de la magnitud del error de predicción se divide en N intervalos, por ejemplo $N = 6$. Si el rango dinámico de la magnitud es $[0, \dots, 2^{q-1}-1]$, donde q es la precisión de cuantificación para la posición de coeficiente dada, (tal como se especifica en las matrices anteriores) entonces cada intervalo n, $n = 0, \dots, N-1$, puede abarcar los siguientes rangos:

45

$$[n_{start}, n_{end}] = [0, \dots, 2^{q-N}] \quad \text{if } n = 0$$

$$[n_{start}, n_{end}] = [2^{q-N+n-1}, 2^{q-N+n}] \quad \text{if } n > 0$$

5 En este caso, el intervalo b , $b = 0, \dots, N-1$, al que pertenece la magnitud de entrada m , puede codificarse utilizando un código de prefijo unario (que toma $b+1$ bits). Entonces, el resto de la magnitud, $m - b_{inicio}$, puede codificarse usando el código de sufijo de longitud fija de $(q - N + b - 1)$ bits. Finalmente, el signo del error de predicción se codifica utilizando 1 bit.

10 Por ejemplo, para el coeficiente central en el filtro 1D, la unidad de predicción y cuantificación de coeficientes 41 puede utilizar una precisión de 9 bits para cuantificar el error de predicción, es decir, $q = 9$, de los cuales 8 bits pueden usarse para cuantificar la magnitud del error. Si la magnitud del error es 97, entonces está dentro del 4º intervalo, es decir, $b = 4$. Por lo tanto, el código de prefijo unario "11110" puede codificarse para indicar el cuarto intervalo, y el resto de $m - b_{inicio} = 97 - 64 = 33$ puede codificarse usando un código de sufijo de longitud fija de $(q - N + b - 1) = (9 - 6 + 4 - 1) = 6$ bits. Finalmente, se puede codificar un bit "0" para indicar el signo positivo.

15 Las técnicas de esta divulgación se pueden implementar en una gran variedad de dispositivos o aparatos, incluyendo un teléfono inalámbrico, un circuito integrado (IC) o un conjunto de IC (es decir, un conjunto de chips). Se ha descrito cualquier componente, módulo o unidad para enfatizar aspectos funcionales y no necesariamente requieren realización por diferentes unidades de hardware.

20 Por consiguiente, las técnicas descritas en el presente documento pueden implementarse en hardware, software, firmware o cualquier combinación de lo anterior. Si implementan en hardware, cualquier característica descrita como módulos, unidades o componentes pueden implementarse juntos en un dispositivo lógico integrado o por separado, como dispositivos lógicos discretos pero interoperables. Si se implementan en software, las técnicas pueden realizarse, al menos en parte, mediante un medio legible por ordenador que comprende instrucciones que, al ejecutarse en un procesador, realizan uno o más de los métodos descritos anteriormente. El medio legible por ordenador puede comprender un medio de almacenamiento legible por ordenador y puede formar parte de un producto de programa informático, que puede incluir materiales de empaquetado. El medio de almacenamiento legible por ordenador puede comprender memoria de acceso aleatorio (RAM) tal como memoria de acceso aleatorio dinámica síncrona (SDRAM), memoria de sólo lectura (ROM), memoria de acceso aleatorio no volátil (NVRAM), memoria de sólo lectura programable y borrrable eléctricamente (EEPROM), memoria flash, medios de almacenamiento de datos magnéticos u ópticos, y similares. Las técnicas se pueden realizar adicionalmente, o alternativamente, al menos en parte por un medio de comunicación legible por ordenador que lleva o comunica código en forma de instrucciones o estructuras de datos y que se puede acceder, leer y/o ejecutar por un ordenador.

35 El código puede ser ejecutado por uno o más procesadores, tales como uno o más procesadores de señales digitales (DSP), microprocesadores de propósito general, circuitos integrados de aplicación específica (ASIC), matrices lógicas programables por campo (FPGA), u otro circuito lógico integrado o discreto equivalente. Por consiguiente, el término "procesador", como se utiliza en el presente documento, puede referirse a cualquier estructura anterior o a cualquier otra estructura adecuada para la implementación de las técnicas descritas en el presente documento. Además, en algunos aspectos, la funcionalidad descrita en el presente documento puede proporcionarse dentro de módulos de software y/o módulos de hardware dedicados configurados para la codificación y la decodificación, o incorporarse en un códec de vídeo combinado. Además, las técnicas podrían implementarse completamente en uno o más circuitos o elementos lógicos.

45 Se han descrito diversos aspectos de la divulgación. Estos y otros aspectos están dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un método para realizar un proceso de compensación de movimiento durante la codificación o decodificación de un bloque de vídeo dentro de una trama codificada de una secuencia de vídeo, comprendiendo el método:

obtener bloques de píxeles desde una o más tramas de vídeo de referencia en el proceso de compensación de movimiento, en el que los bloques de píxeles comprenden bloques de referencia reconstruidos que se decodificaron previamente e incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro de los bloques de píxeles;

aplicar un filtrado a un valor de píxel entero de los valores de píxeles enteros basándose en otros valores de píxeles enteros dentro de los bloques de píxeles y un valor de píxel entero original del valor de píxel entero para generar un valor de píxel entero ajustado, en el que el valor de píxel entero ajustado corresponde a una posición de píxel entero rodeada por otras posiciones de píxeles enteros a las que corresponden los otros valores de píxeles enteros, y generar un bloque de predicción en el proceso de compensación de movimiento durante la codificación o decodificación del bloque de vídeo basado, al menos en parte, en el valor de píxel entero ajustado;

estando el método **caracterizado por que** un filtro de interpolación de 5 por 5 en forma de

$$p^{FP} = \sum_{i=0}^4 \sum_{j=0}^4 P_{i,j} h_{i,j}^{FP} + o^{FP}$$

se usa para filtrar el píxel entero, en el que p^{FP} es el valor de píxel entero ajustado, $P_{i,j}$ es el valor de píxel entero en la posición (i, j) , $h_{i,j}^{FP}$ es el coeficiente para la posición (i, j) , y o^{FP} es un desfase de filtro.

2. El método de la reivindicación 1, en el que el método se realiza durante un proceso de codificación, comprendiendo además el método generar un bloque de vídeo residual como una diferencia entre un bloque de vídeo actual que está siendo codificado y el bloque de predicción.

3. El método de la reivindicación 1, en el que el método se realiza durante un proceso de codificación, comprendiendo además el método codificar un conjunto de coeficientes de filtro por trama codificada para un filtrado de enteros.

4. El método de la reivindicación 1, en el que el método se realiza durante un proceso de decodificación, comprendiendo además el método generar un bloque de vídeo reconstruido como una suma de un bloque de vídeo residual asociado con un bloque de vídeo actual que está siendo decodificado y el bloque de predicción.

5. El método de la reivindicación 1, en el que el método se realiza durante un proceso de decodificación, comprendiendo además el método recibir un conjunto de coeficientes de filtro por trama codificada para filtrado de enteros.

6. Un dispositivo que realiza un proceso de compensación de movimiento durante la codificación o decodificación de un bloque de vídeo dentro de una trama codificada de una secuencia de vídeo, comprendiendo el dispositivo:

medios para obtener bloques de píxeles desde una o más tramas de vídeo de referencia en el proceso de compensación de movimiento, en el que los bloques de píxeles comprenden bloques de referencia reconstruidos que se decodificaron previamente e incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro del bloque de píxeles;

medios para aplicar un filtrado a un valor de píxel entero de los valores de píxeles enteros basándose en otros valores de píxeles enteros dentro del bloque de píxeles y un valor de píxel entero original del valor de píxel entero para generar un valor de píxel entero ajustado, en el que el valor de píxel entero ajustado corresponde a una posición de píxel entero rodeada por otras posiciones de píxeles enteros a las que corresponden los otros valores de píxeles enteros, y medios para generar un bloque de predicción en el proceso de compensación de movimiento durante la codificación o decodificación del bloque de vídeo basado, al menos en parte, en el valor de píxel entero ajustado; **caracterizado por que**

$$p^{FP} = \sum_{i=0}^4 \sum_{j=0}^4 P_{i,j} h_{i,j}^{FP} + o^{FP}$$

un filtro de interpolación de 5 por 5 en forma de se usa para filtrar el píxel entero, en el que p^{FP} es el valor de píxel entero ajustado, $P_{i,j}$ es el valor de píxel entero en la posición (i, j) , $h_{i,j}^{FP}$ es el coeficiente para la posición (i, j) , y o^{FP} es un desfase de filtro.

7. El dispositivo de la reivindicación 6, en el que el dispositivo realiza un proceso de codificación, comprendiendo el dispositivo medios para generar un bloque de vídeo residual como una diferencia entre un bloque de vídeo actual que está siendo codificado y el bloque de predicción.
- 5 8. El dispositivo de la reivindicación 6, en el que el dispositivo realiza un proceso de decodificación, comprendiendo además el dispositivo medios para generar un bloque de vídeo reconstruido como una suma de un bloque de vídeo residual asociado con un bloque de vídeo actual que está siendo decodificado y el bloque de predicción.
- 10 9. Un medio de almacenamiento no transitorio legible por ordenador que comprende instrucciones que, al ejecutarse por un procesador, hacen que el procesador realice un proceso de compensación de movimiento durante la codificación o decodificación de un bloque de vídeo dentro de una trama codificada de una secuencia de vídeo, en el que las instrucciones hacen que el procesador:

15 obtenga bloques de píxeles desde una o más tramas de vídeo de referencia en el proceso de compensación de movimiento, en el que los bloques de píxeles comprenden bloques de referencia reconstruidos que se decodificaron previamente e incluyen valores de píxeles enteros correspondientes a posiciones de píxeles enteros dentro del bloque de píxeles;

20 aplique un filtrado a un valor de píxel entero de los valores de píxeles enteros basándose en otros valores de píxeles enteros dentro del bloque de píxeles y un valor de píxel entero original del valor de píxel entero para generar un valor de píxel entero ajustado, en el que el valor de píxel entero ajustado corresponde a una posición de píxel entero rodeada por otras posiciones de píxeles enteros a las que corresponden los otros valores de píxeles enteros, y genere un bloque de predicción en el proceso de compensación de movimiento durante la codificación o decodificación del bloque de vídeo basado, al menos en parte, en el valor de píxel entero ajustado;

25

caracterizado por que

$$p^{FP} = \sum_{i=0}^4 \sum_{j=0}^4 P_{i,j} h_{i,j}^{FP} + o^{FP}$$

30 un filtro de interpolación de 5 por 5 en forma de p^{FP} se usa para filtrar el píxel entero, en el que p^{FP} es el valor de píxel entero ajustado, $P_{i,j}$ es el valor de píxel entero en la posición (i, j), $h_{i,j}^{FP}$ es el coeficiente para la posición (i, j), y o^{FP} es un desfase de filtro.

10. El medio de almacenamiento no transitorio legible por ordenador de la reivindicación 9, en el que las instrucciones hacen que el dispositivo realice un proceso de decodificación, en el que las instrucciones hacen que el dispositivo genere un bloque de vídeo reconstruido como una suma de un bloque de vídeo residual asociado con un bloque de vídeo actual que está siendo decodificado y el bloque de predicción.
- 35

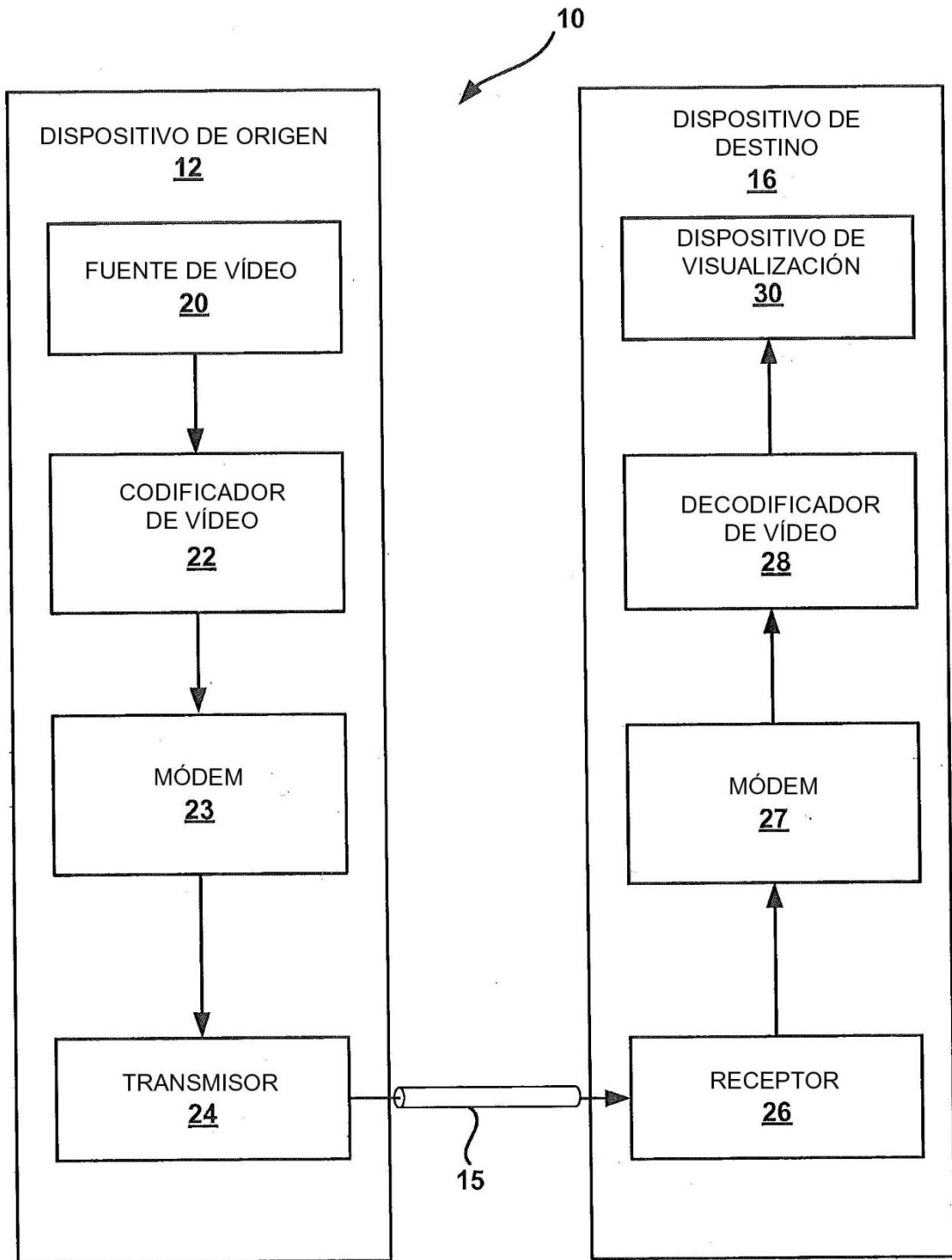


FIG. 1

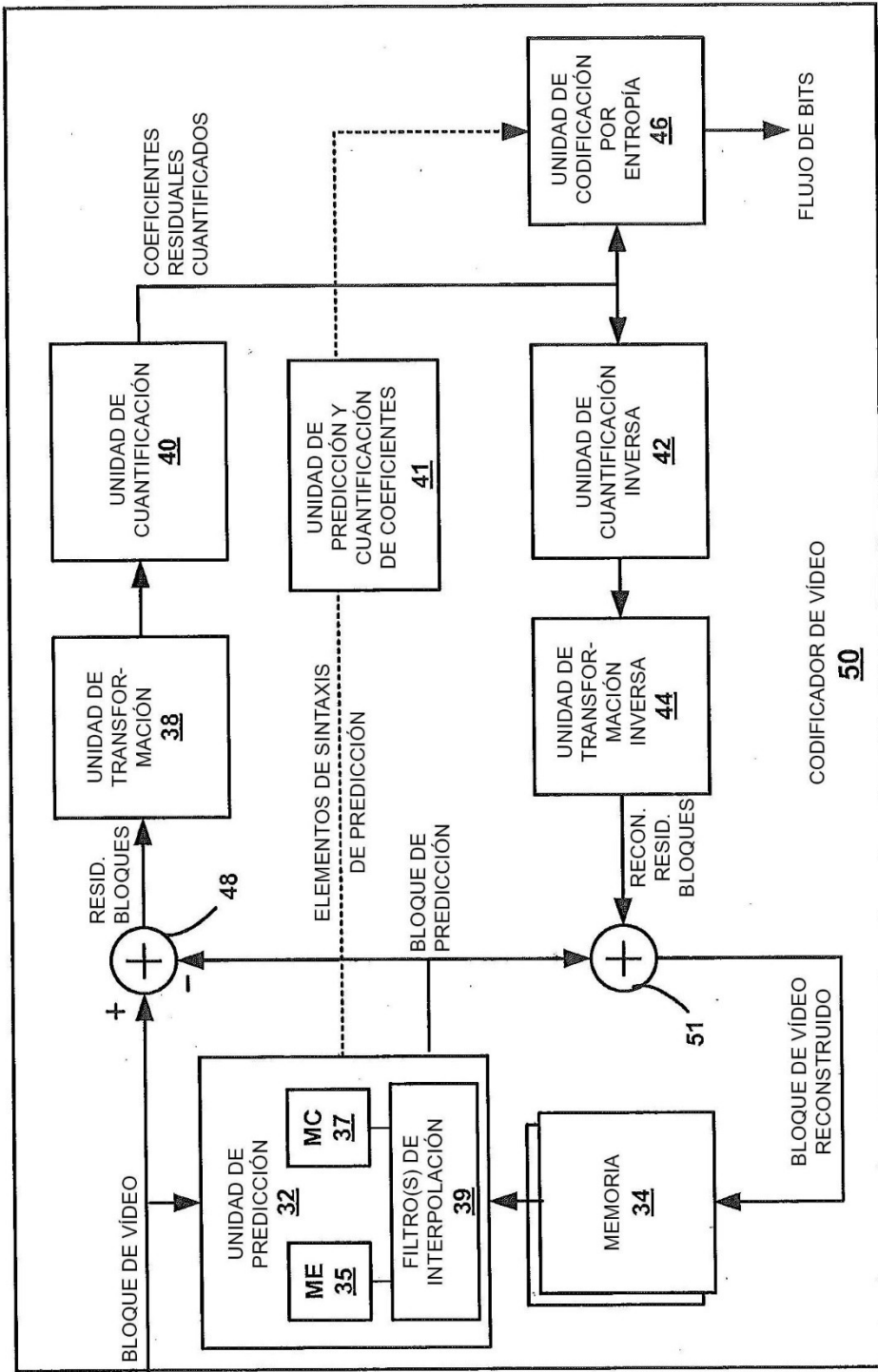


FIG. 2

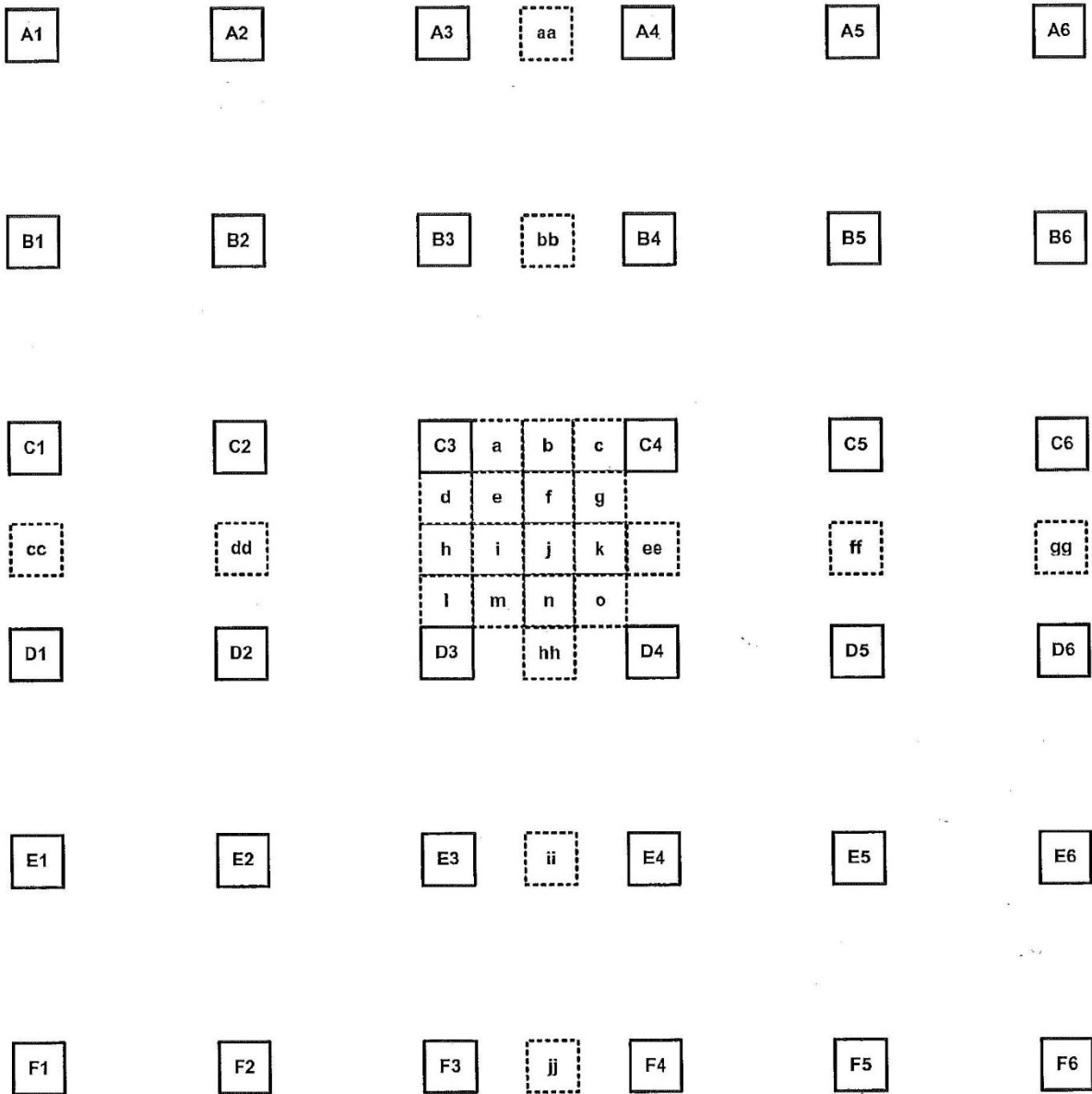


FIG. 3

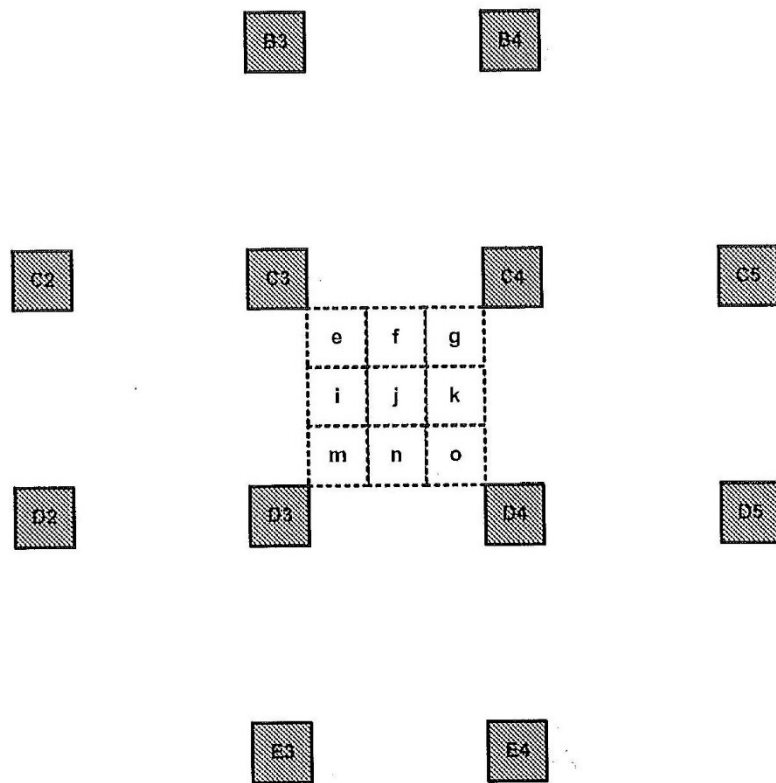


FIG. 4

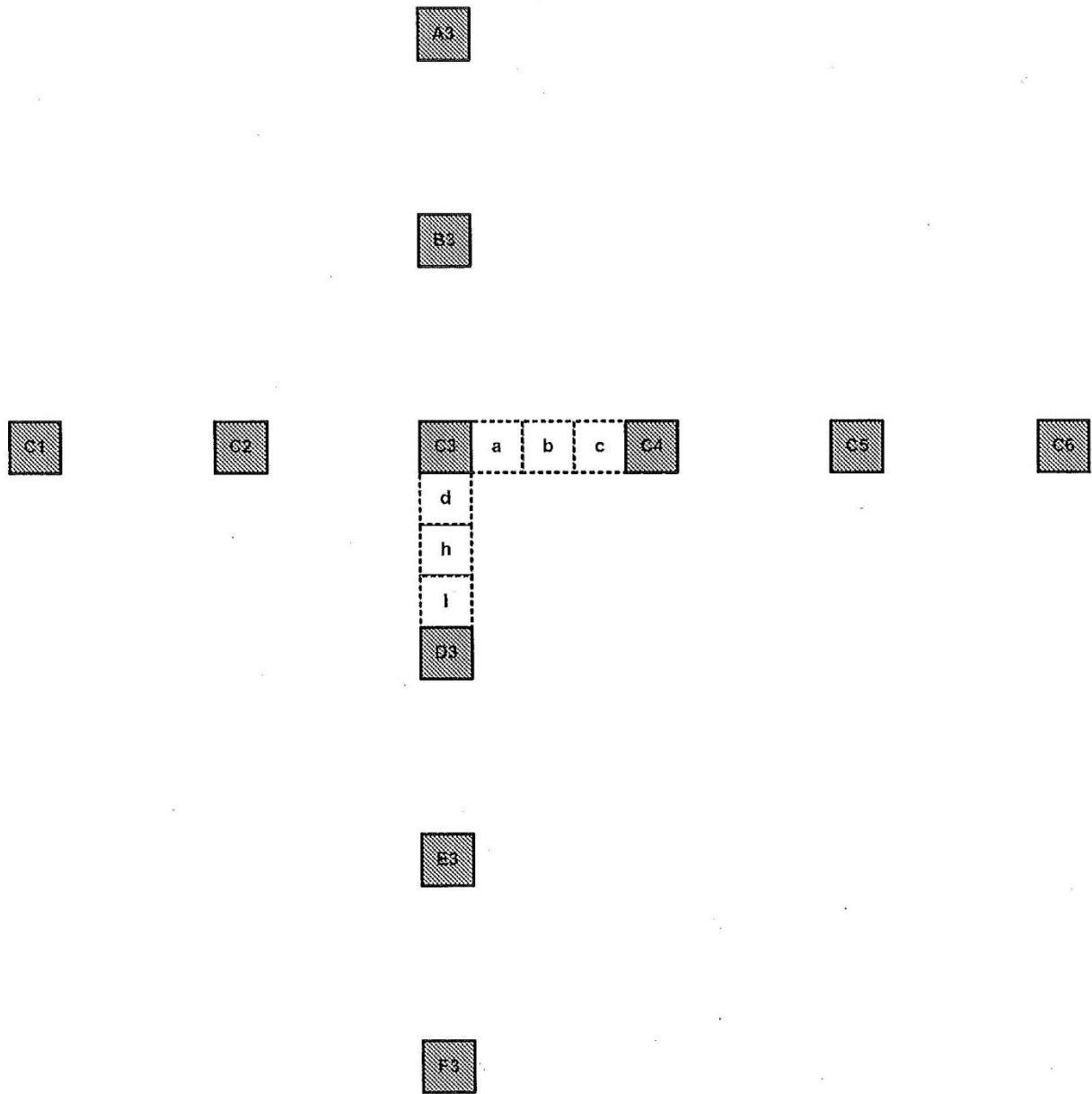


FIG. 5

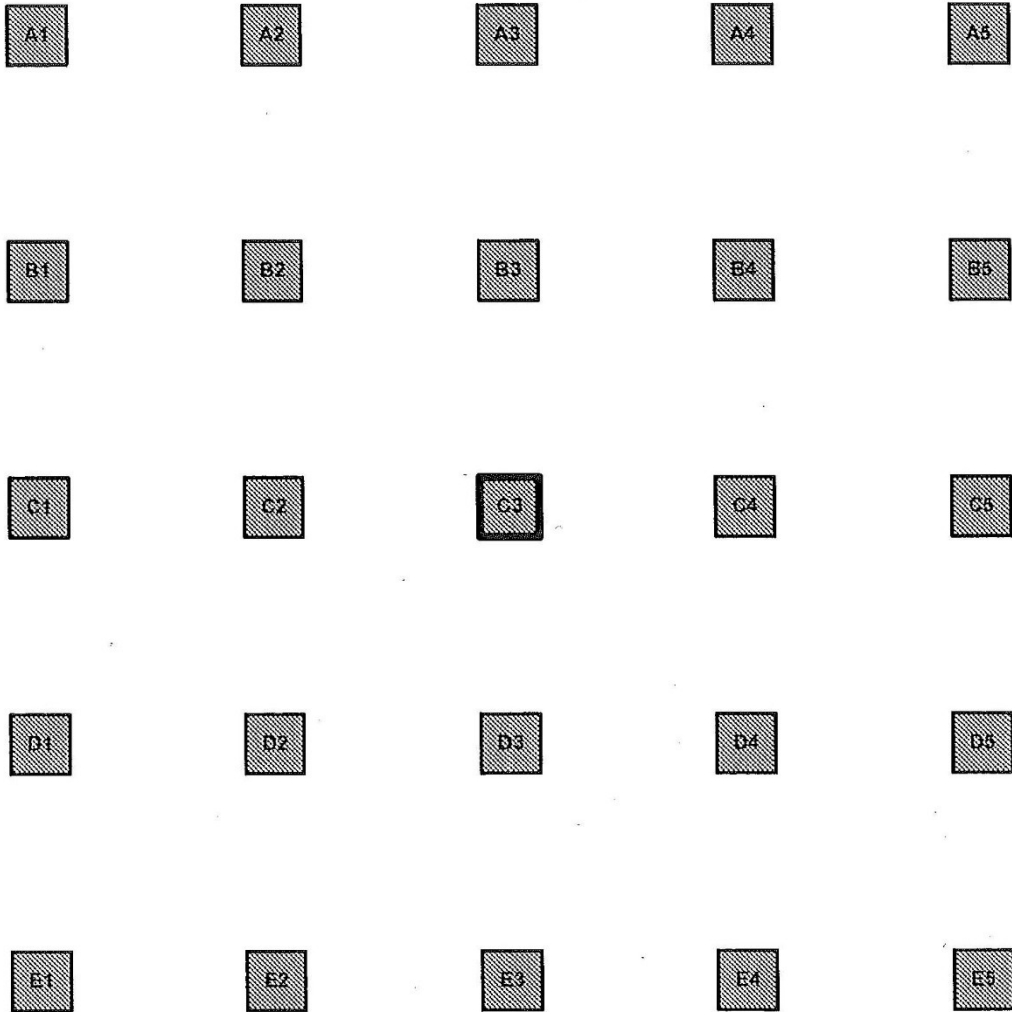


FIG. 6

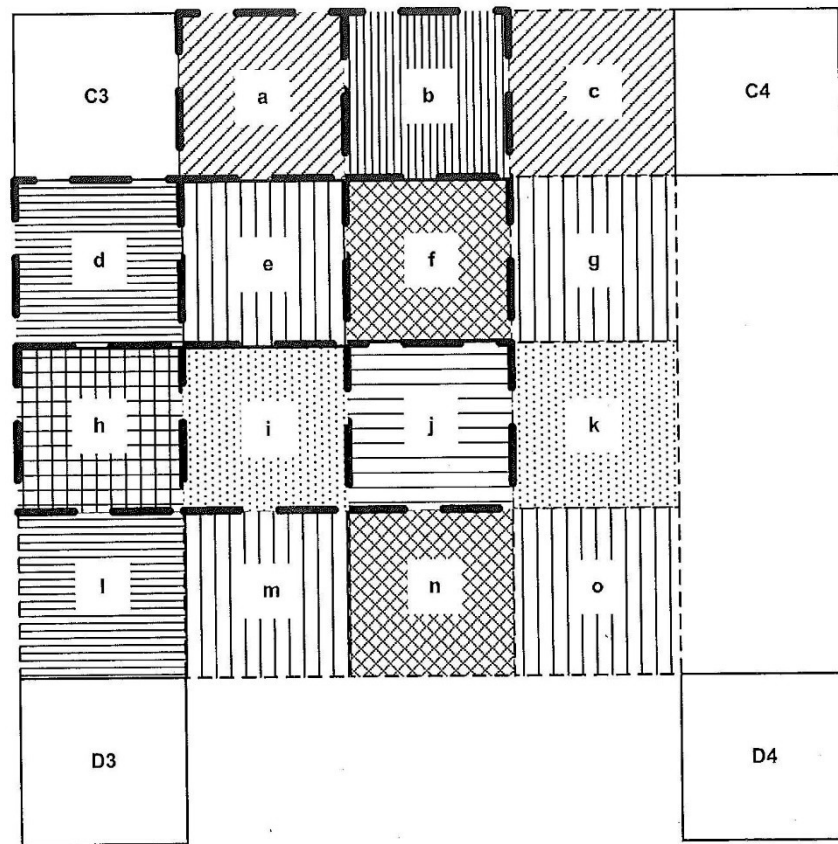


FIG. 7



FIG. 8



FIG. 9



FIG. 10



FIG. 11

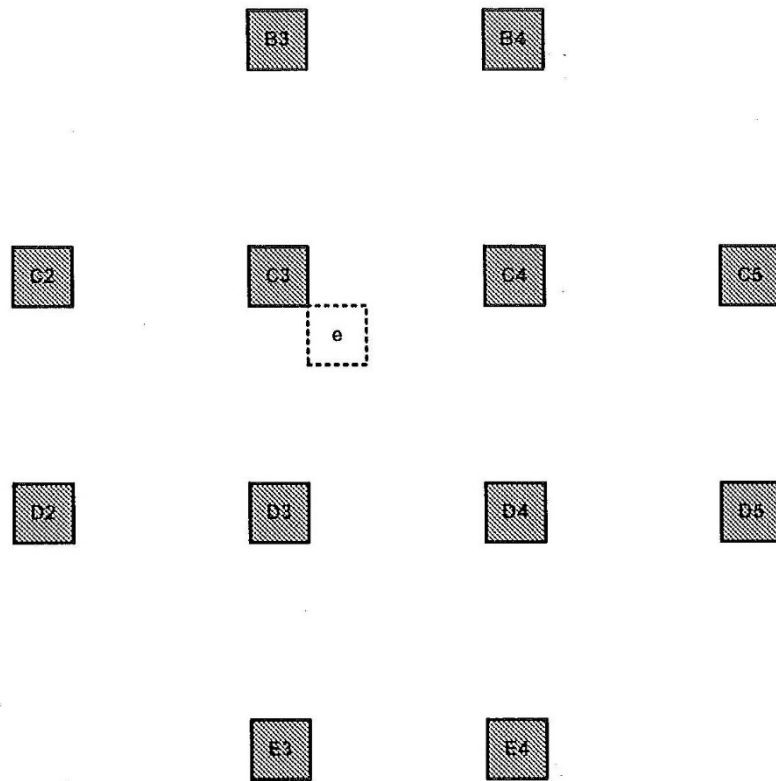


FIG. 12

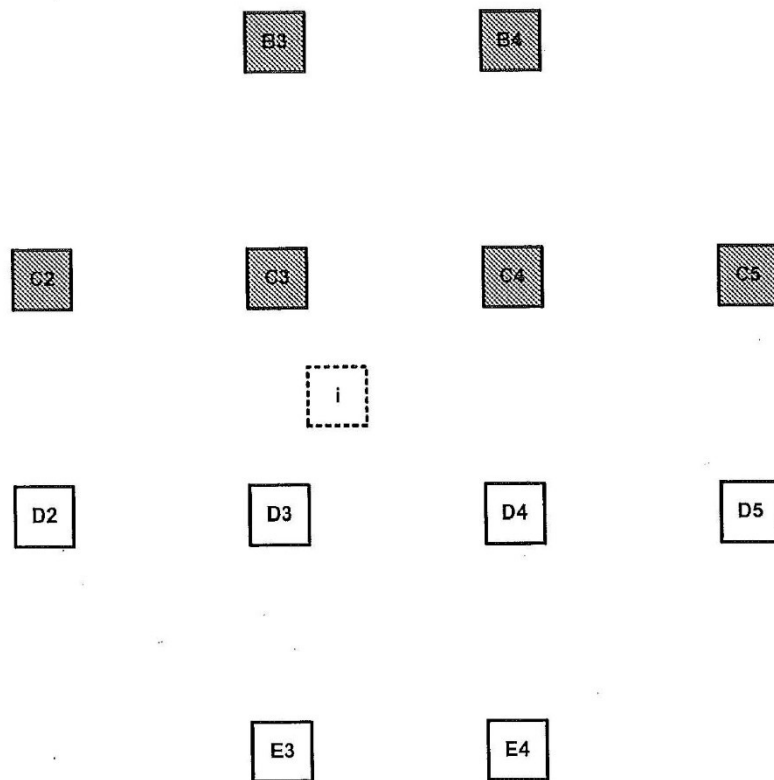


FIG. 13

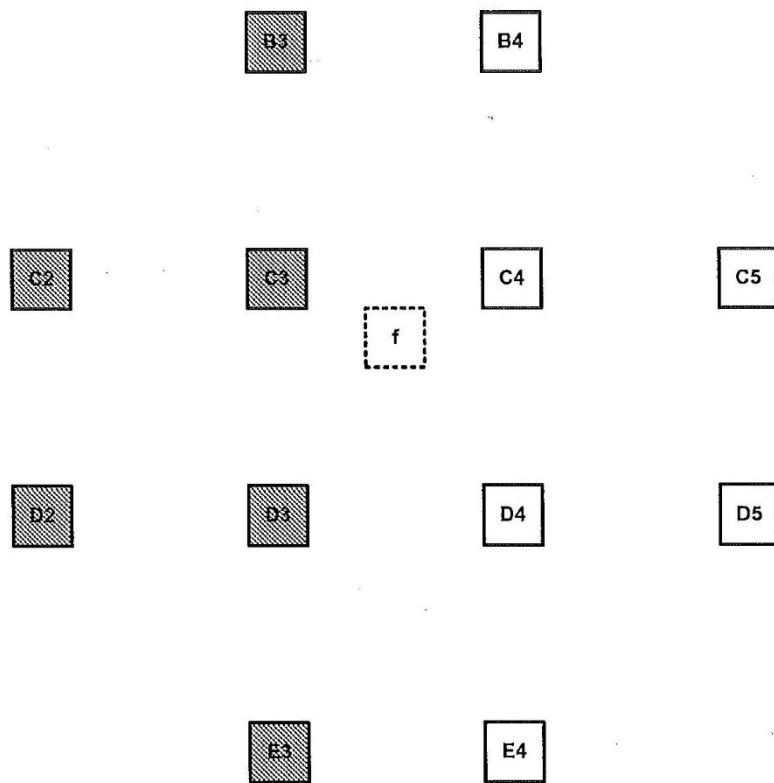


FIG. 14

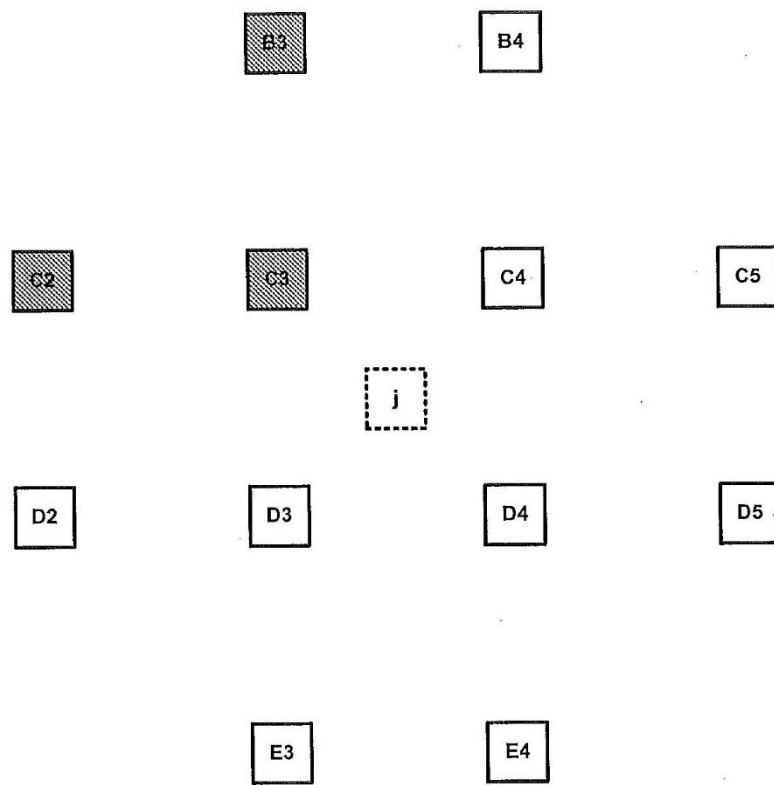


FIG. 15

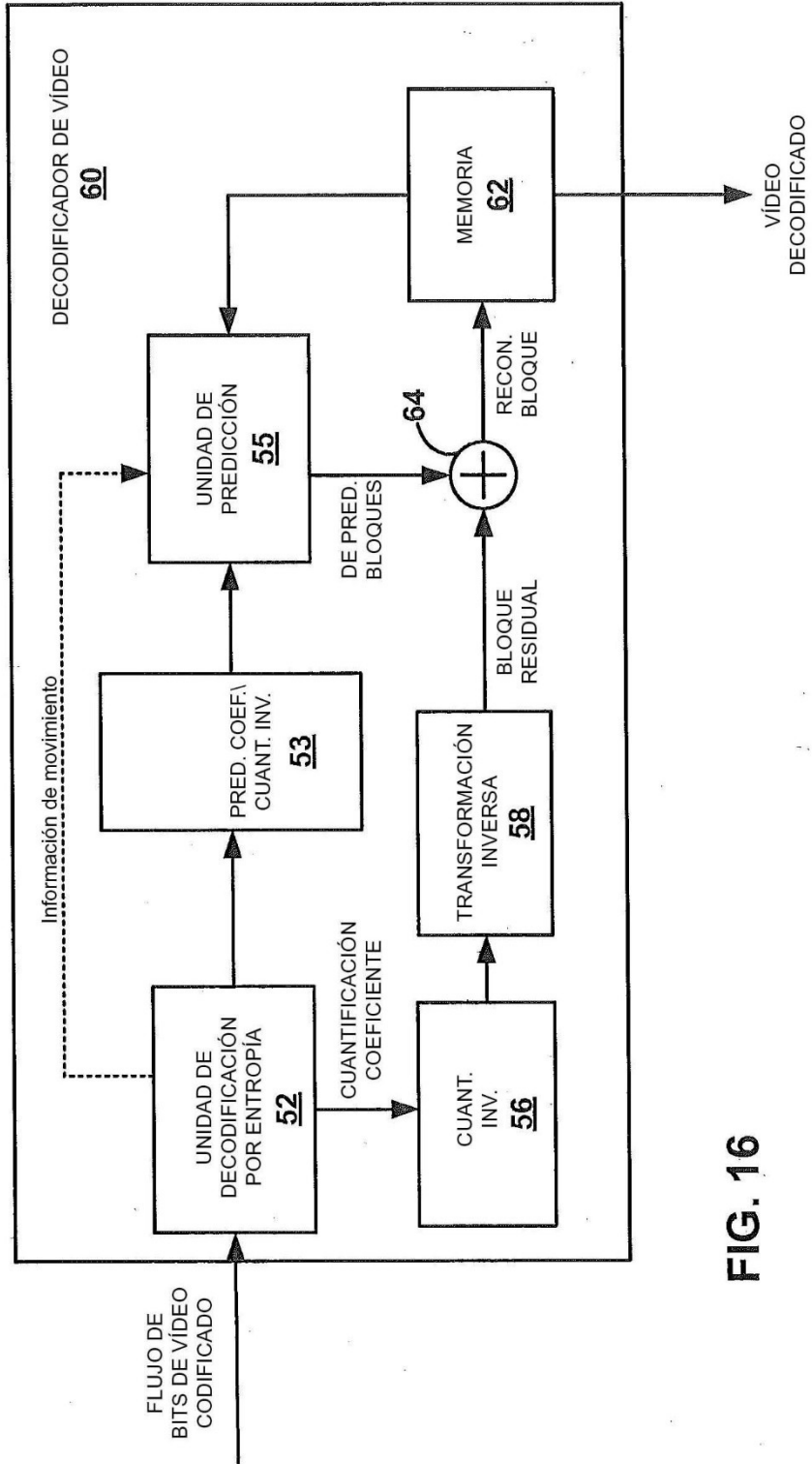


FIG. 16

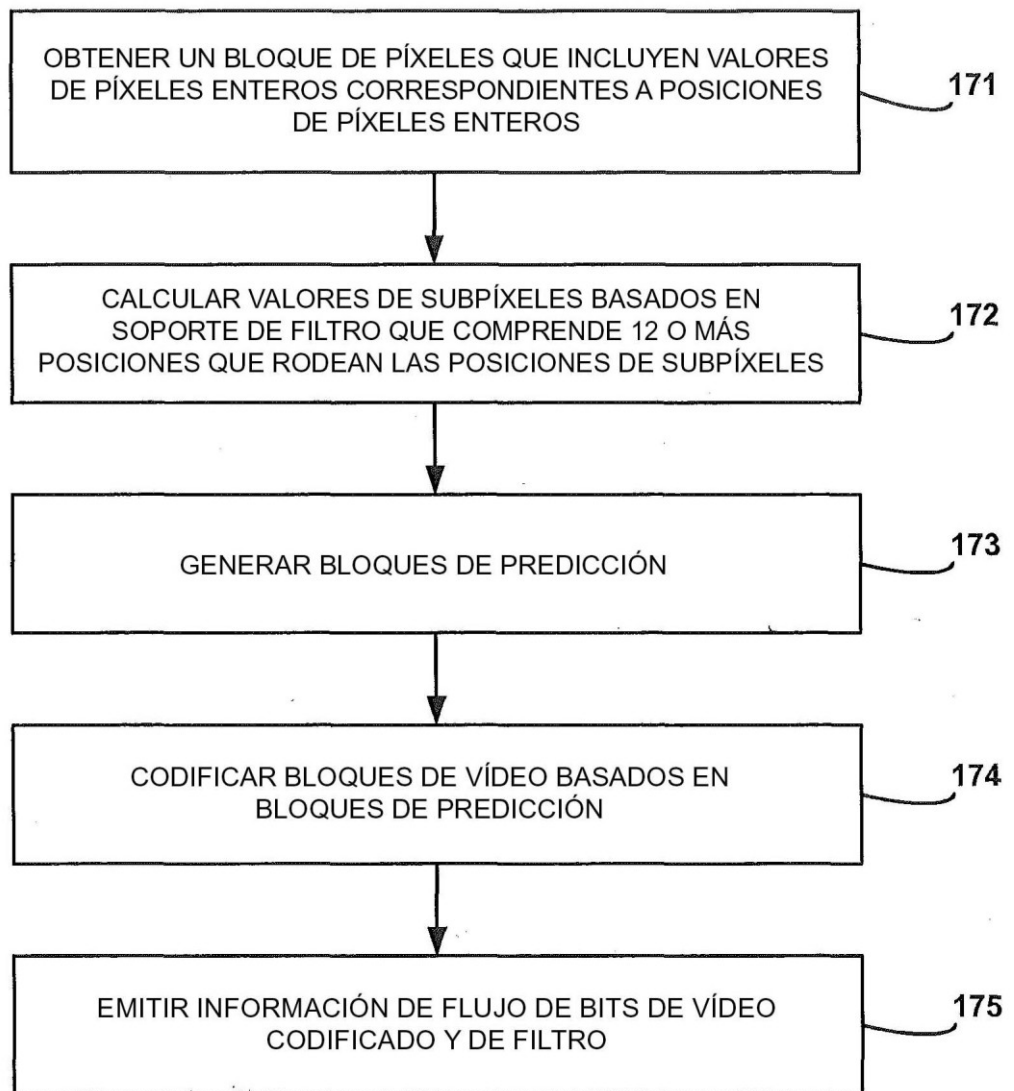


FIG. 17

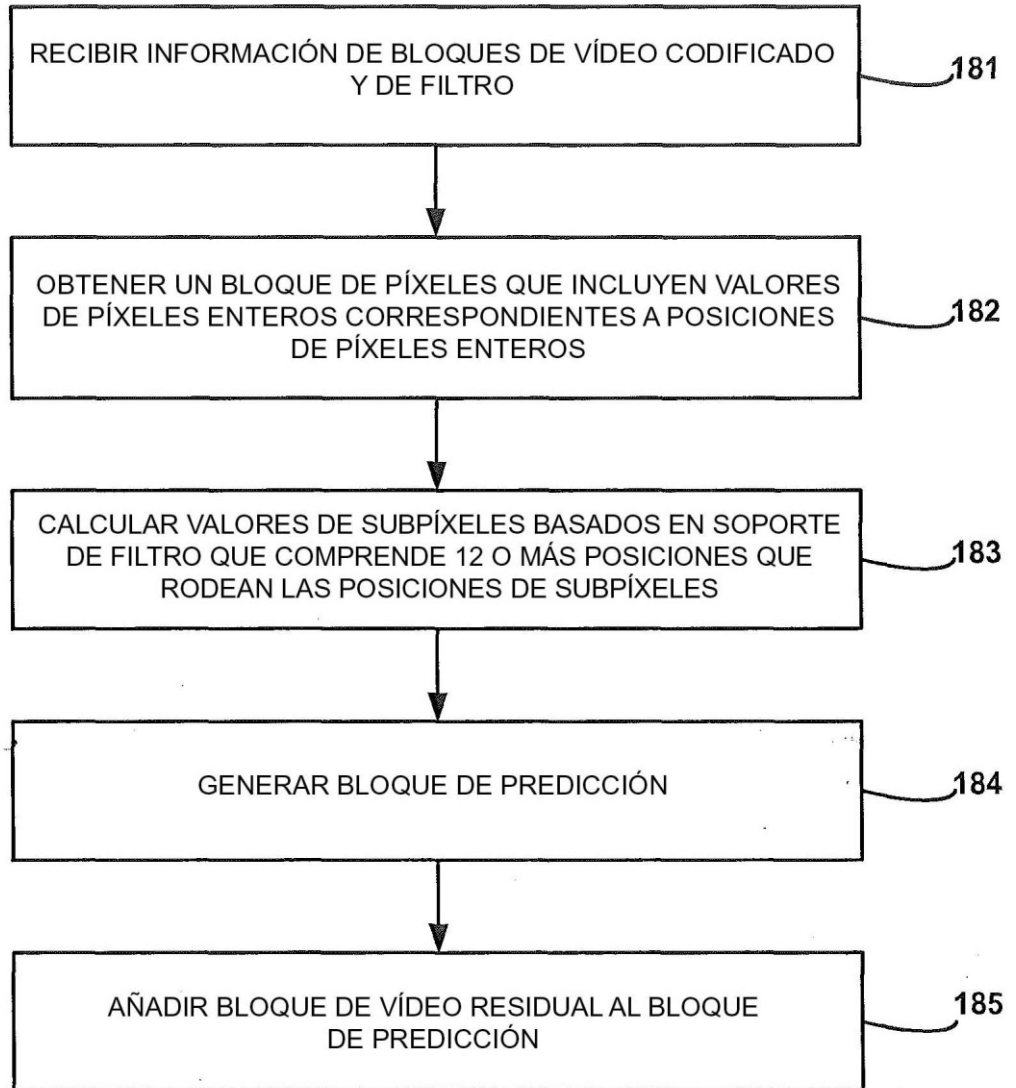


FIG. 18

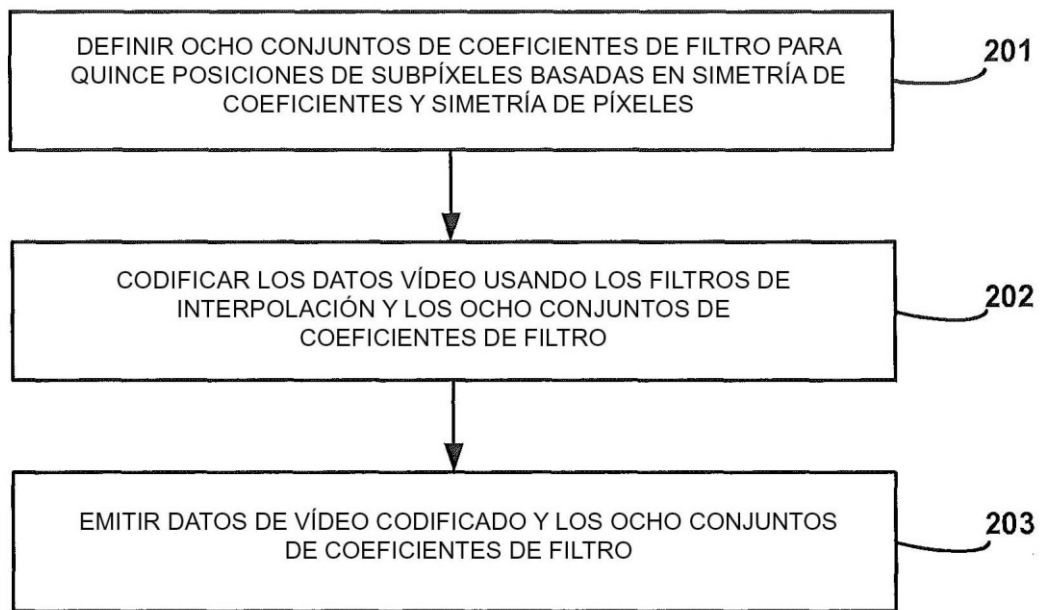


FIG. 19

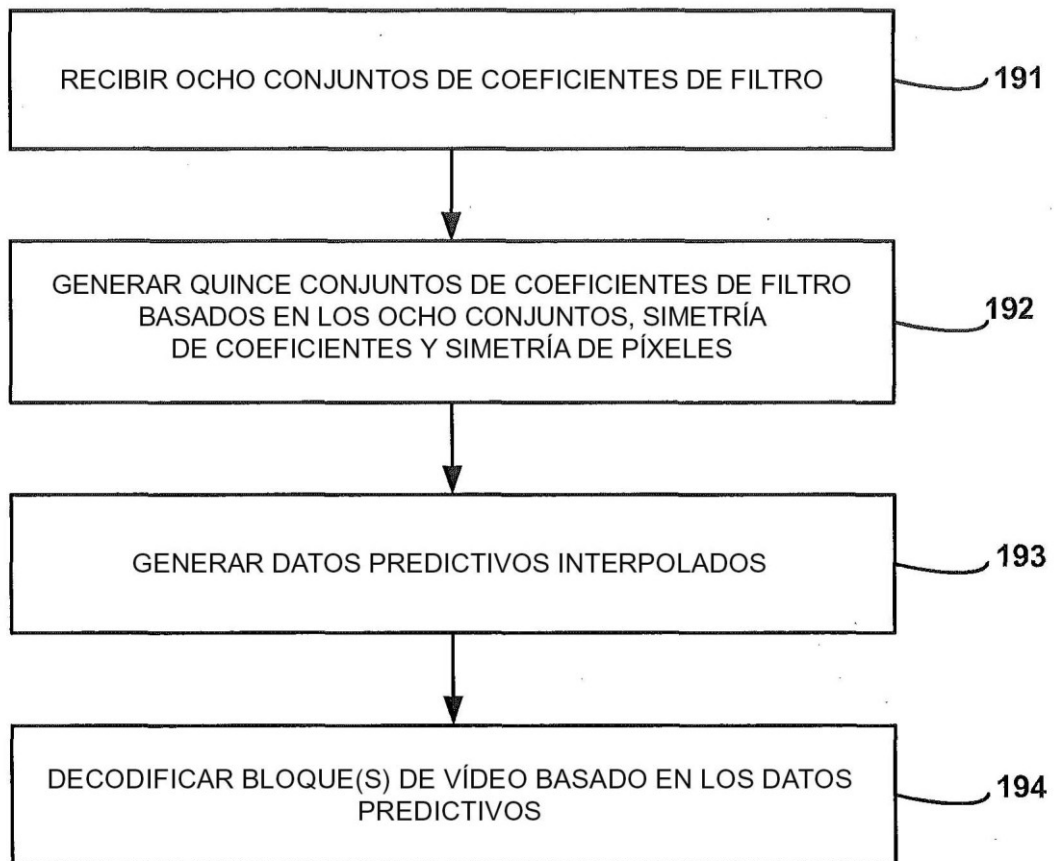


FIG. 20

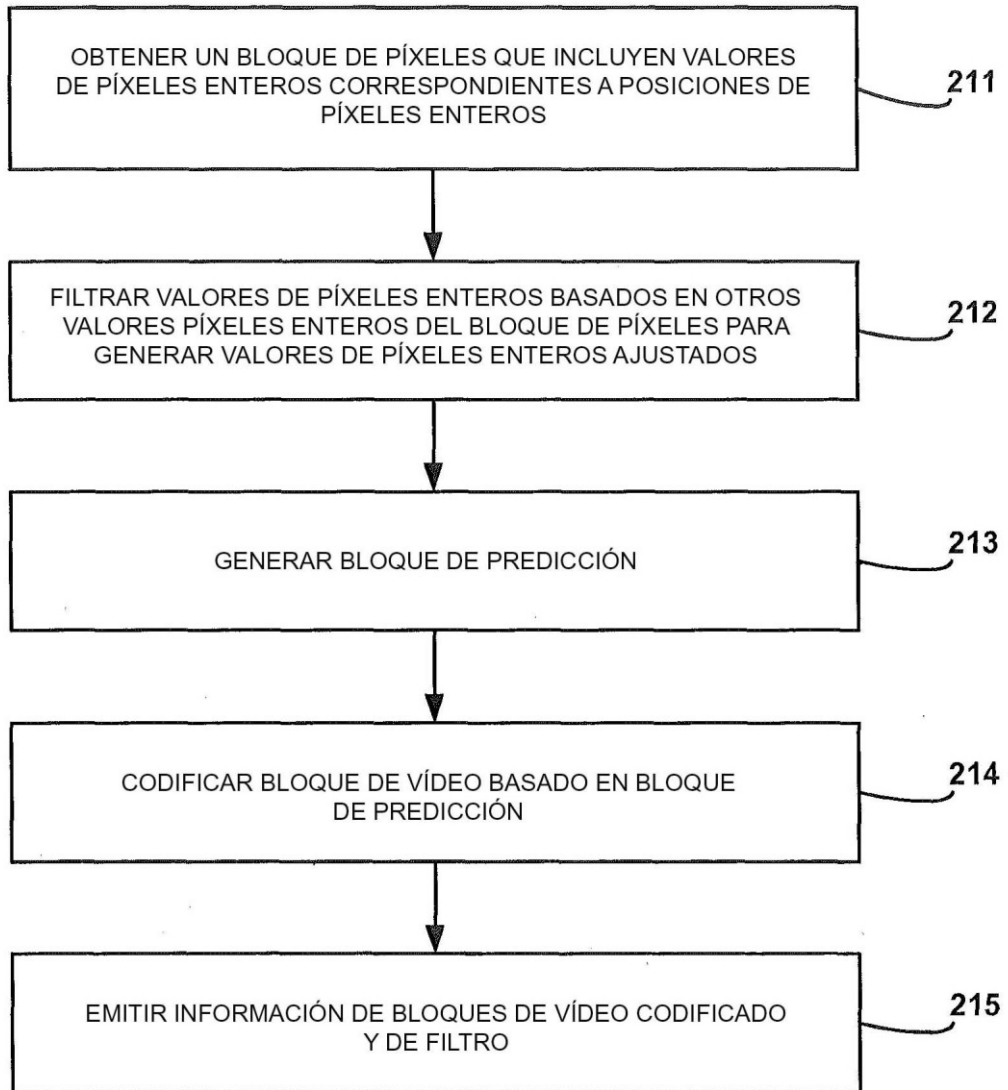


FIG. 21

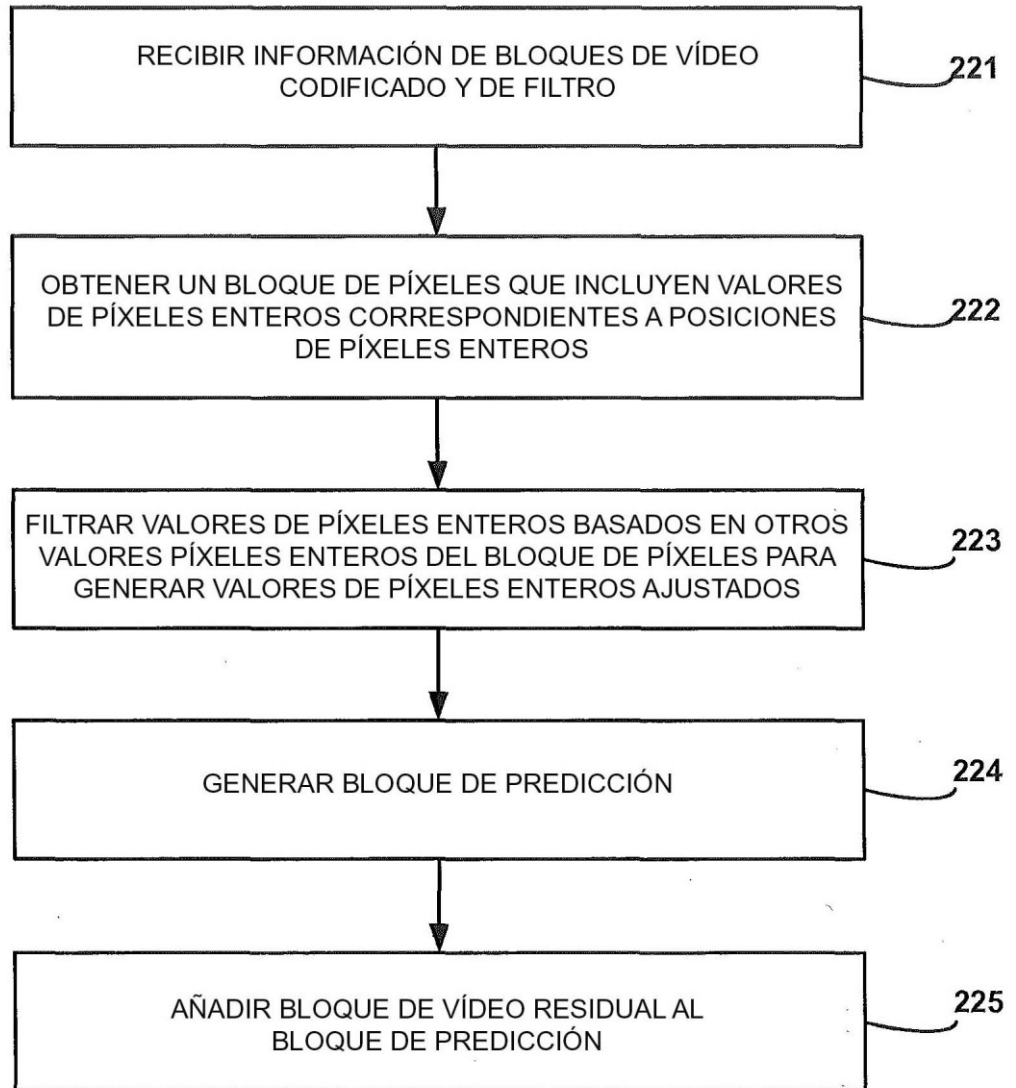


FIG. 22

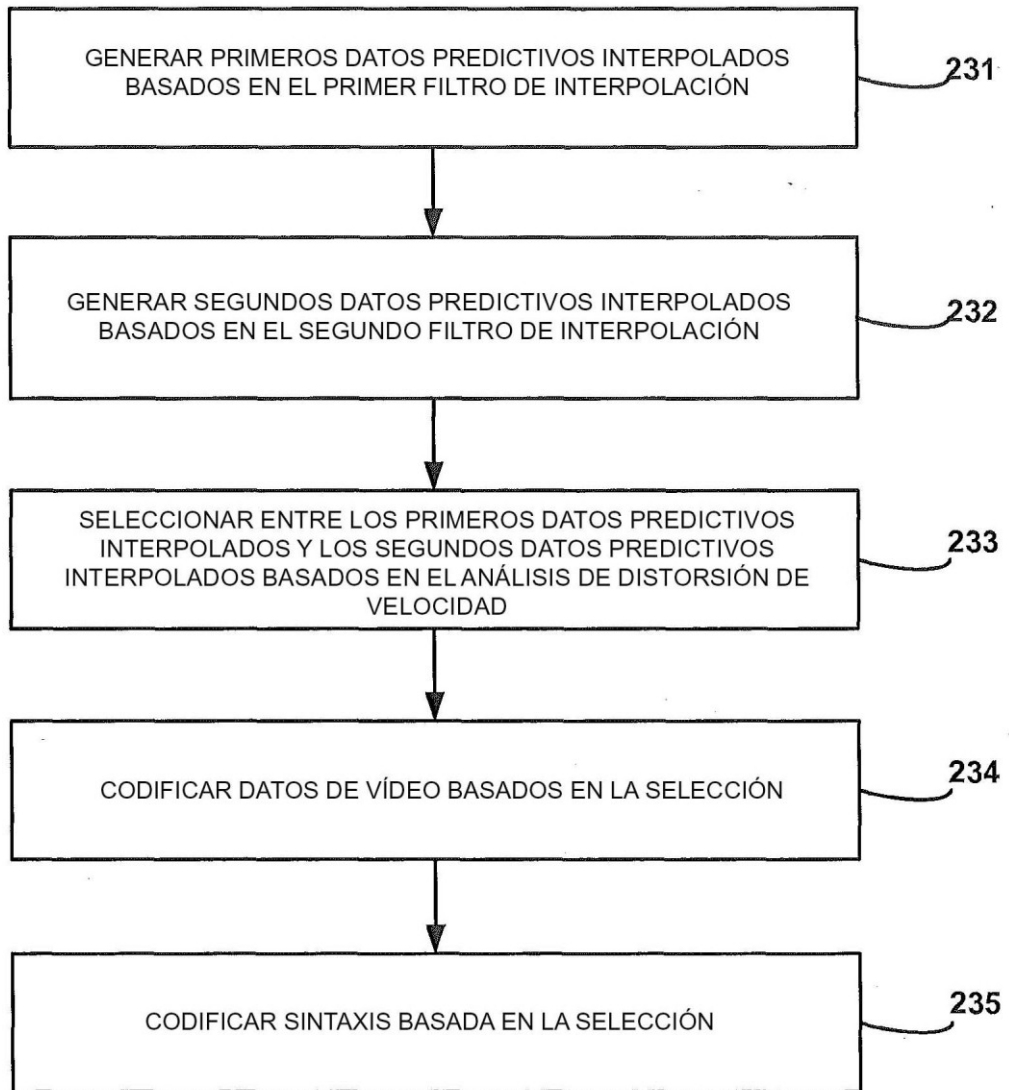


FIG. 23

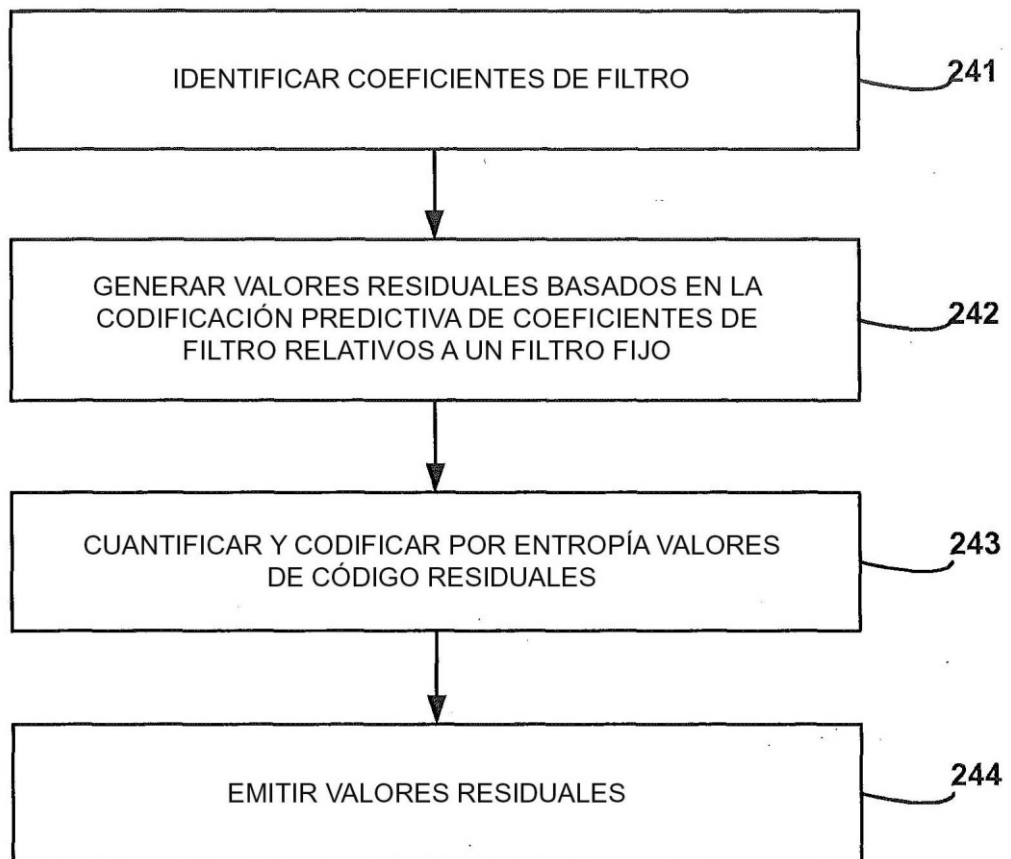


FIG. 24

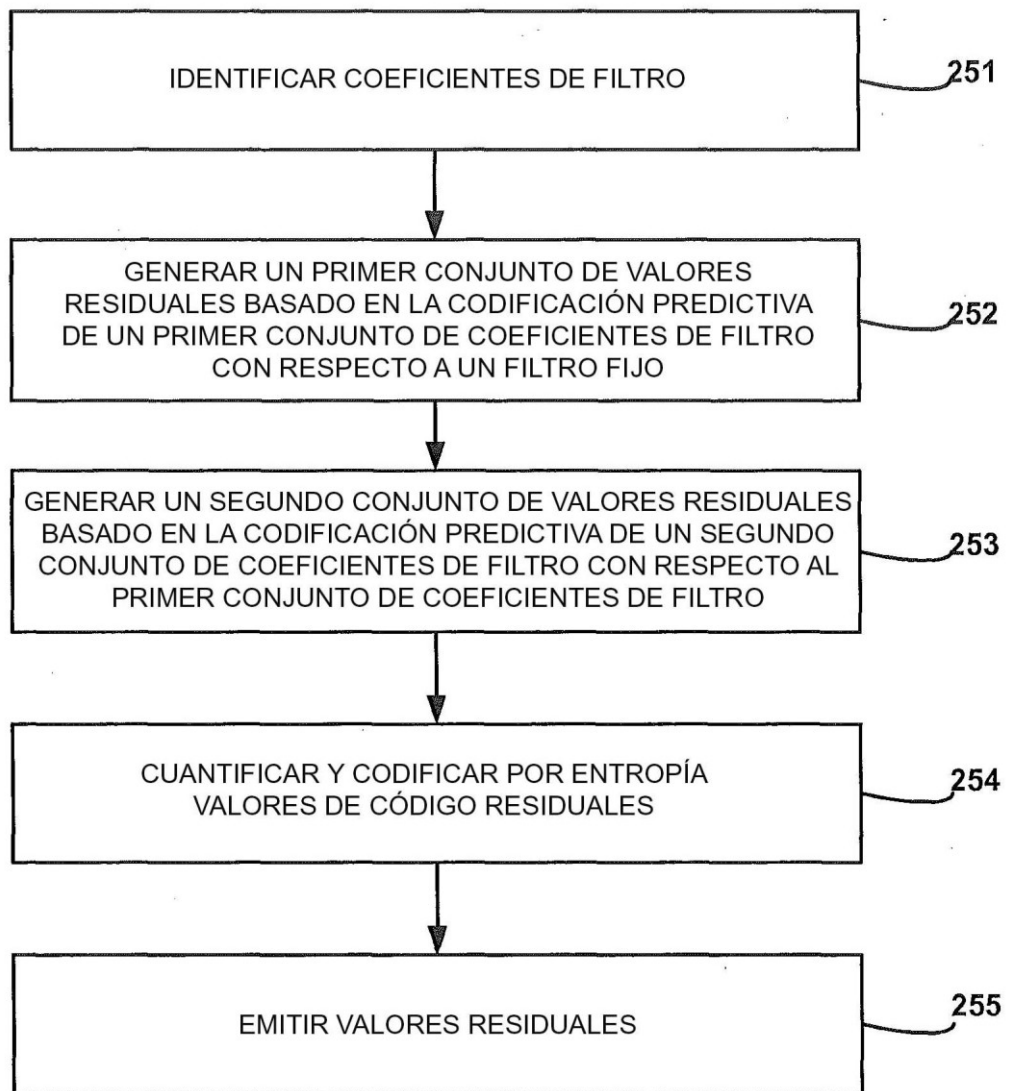


FIG. 25

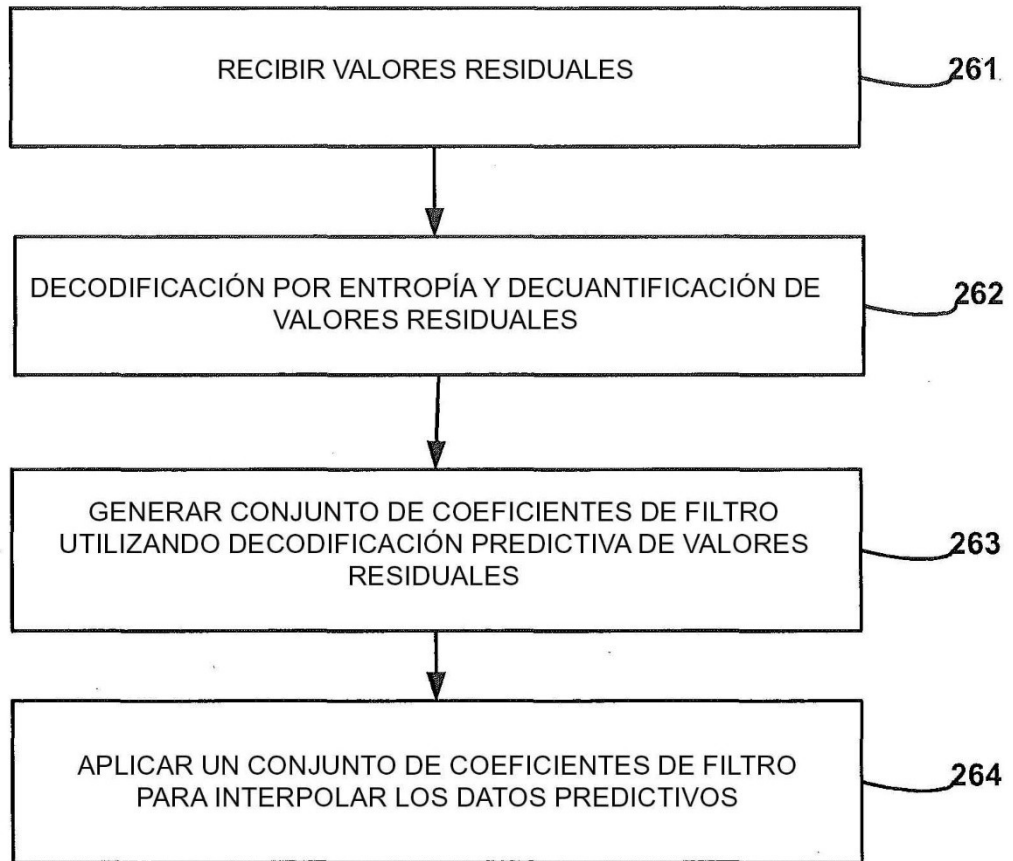


FIG. 26

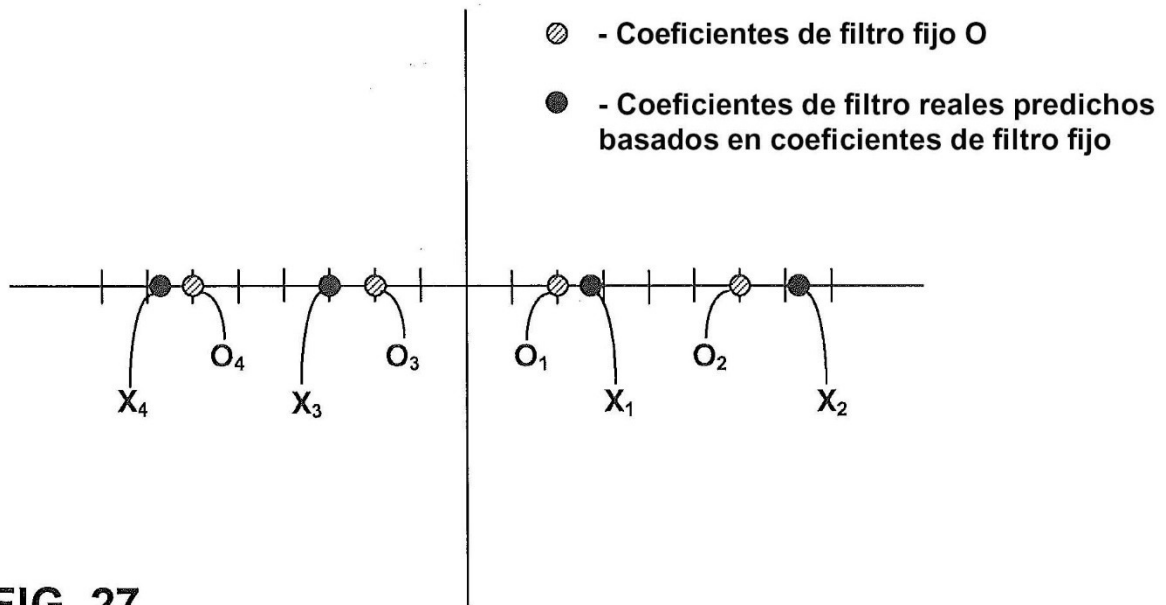


FIG. 27

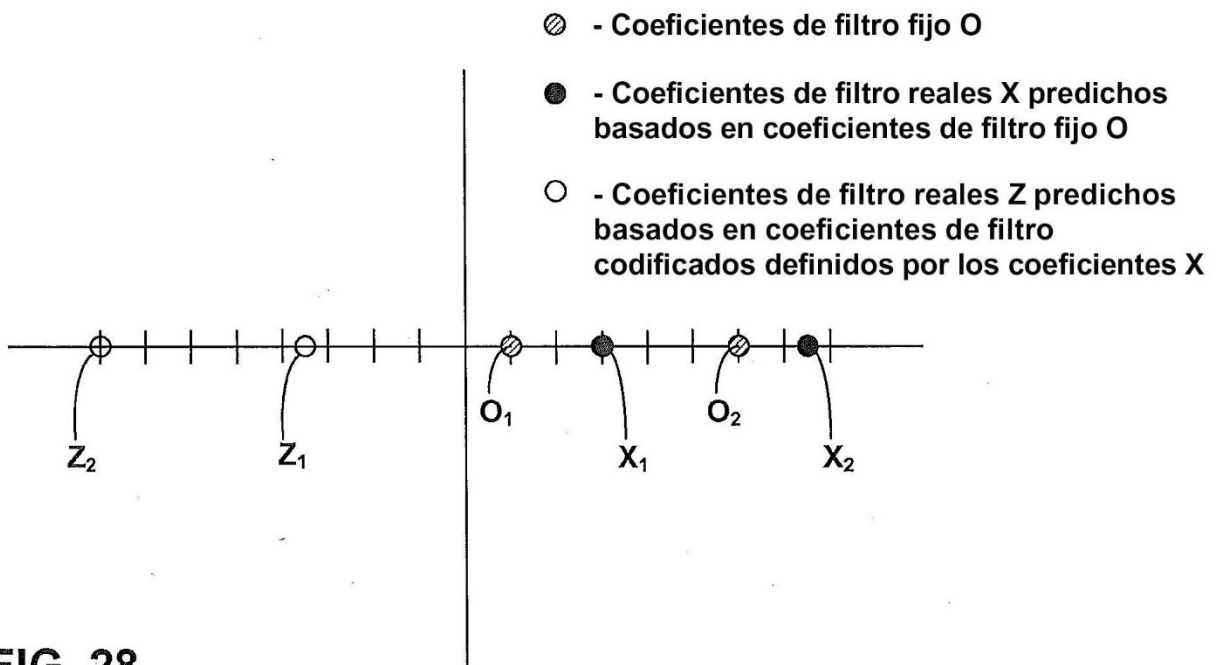


FIG. 28

h0, 0	h0, 1	h0, 2	h0, 3	h0, 4
h1, 0	h1, 1	h1, 2	h1, 3	h1, 4
h2, 0	h2, 1	h2, 2	h2, 3	h2, 4
h3, 0	h3, 1	h3, 2	h3, 3	h3, 4
h4, 0	h4, 1	h4, 2	h4, 3	h4, 4

FIG. 29