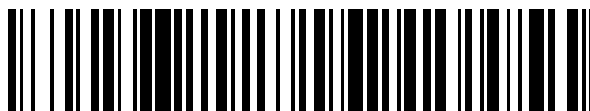


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 441 223**

51 Int. Cl.:

H04N 7/36

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **08.01.2003 E 10169924 (7)**

97 Fecha y número de publicación de la concesión europea: **16.10.2013 EP 2249576**

54 Título: **Procedimiento de decodificación de imágenes y aparato de decodificación de imágenes**

30 Prioridad:

09.01.2002 JP 2002001983

12.07.2002 JP 2002204714

28.11.2002 JP 2002346062

45 Fecha de publicación y mención en BOPI de la
traducción de la patente:
03.02.2014

73 Titular/es:

PANASONIC CORPORATION (100.0%)

1006, Oaza Kadoma Kadoma-shi

Osaka 571-8501, JP

72 Inventor/es:

KONDO, SATOSHI;

ABE, KIYOFUMI;

HAGAI, MAKOTO y

KADONO SHINYA

74 Agente/Representante:

ROEB DÍAZ-ÁLVAREZ, María

ES 2 441 223 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

Procedimiento de decodificación de imágenes y aparato de decodificación de imágenes

5 Campo técnico

La presente invención se refiere a un procedimiento de codificación de vectores de movimiento y un procedimiento de decodificación de vectores de movimiento usando codificación de predicción entre imágenes.

10 Técnica anterior

En la era multimedia que maneja de forma integral audio, vídeo y otra información, los medios de información existentes, es decir, periódicos, revistas, televisiones, radios, teléfonos y otros medios a través de los cuales se transmite la información a las personas, se han incluido recientemente en el ámbito multimedia. En general, multimedia se refiere a algo que se representa asociando no sólo caracteres, sino también gráficos, voz y especialmente imágenes y similares conjuntamente, pero con el fin de incluir los medios de información existentes mencionados anteriormente en el ámbito multimedia, parece que un requisito previo es representar dicha información en forma digital.

Sin embargo, cuando se calcula la cantidad de información contenida en cada uno de los medios de información mencionados anteriormente en forma de cantidad de información digital, mientras que la cantidad de información por carácter es de 1~2 bytes, la cantidad de información que se requerirá para voz es de 64 Kbits o más por segundo (calidad telefónica), y 100 Mbits o más por segundo para imágenes en movimiento (calidad actual de recepción en televisión), y no es realista que los medios de información mencionados anteriormente manejen una cantidad de información tan enorme como ésta en forma digital. Por ejemplo, aunque los videoteléfonos están ya en uso real por medio de la Red Digital de Servicios Integrados (RDSI) que ofrece una velocidad de transmisión de 64 Kbps/s~1,5 Mbps/s, no es práctico emitir tomas de vídeo por cámaras de televisión directamente a través de RDSI.

Sobre este telón de fondo, han empezado a requerirse técnicas de compresión de información, y se emplean técnicas de compresión de imágenes en movimiento que cumplen con las normas H.261 y H.263 normalizadas internacionalmente por la ITU-T (Unión Internacional de Telecomunicación-Sector de Normalización de Telecomunicación) para videoteléfonos, por ejemplo (Véase, por ejemplo, Information technology – Coding of audio-visual objects - Part 2: video (ISO/IEC 14496-2), pág. 146-148, 1999.12.1). Por otra parte, según técnicas de compresión de información que cumplen con la norma MPEG-1, es posible almacenar información de imágenes en un CD (disco compacto) de música corriente junto con información sonora.

Aquí, MPEG (Moving Picture Experts Group) es una norma internacional sobre compresión de señales de imágenes en movimiento, y MPEG-1 es una norma para compresión de información de señales de televisión aproximadamente en una centésima parte de manera que las señales de imágenes en movimiento pueden emitirse a una velocidad de 1,5 Mbps. Además, como la velocidad de transmisión dentro del ámbito de la norma MPEG-1 está limitada principalmente a 1,5 Mbps aproximadamente, MPEG-2, que fue normalizada con vistas a satisfacer los requisitos de una calidad de imagen mejorada, permite la transmisión de datos de señales de imágenes en movimiento a una velocidad de 2~15 Mbps. Además, MPEG-4 que consigue una relación de compresión superior que la de MPEG-1 y MPEG-2, permite la codificación, decodificación y operación en una unidad objeto, y realiza un nuevo funcionamiento requerido para la era multimedia, ha sido normalizada por el grupo de trabajo (ISO/IEC JTC1/SC29/WG11) que se ha comprometido en la normalización de MPEG-1 y MPEG-2. MPEG-4 se dirigió inicialmente a la normalización de un procedimiento de codificación para una velocidad binaria baja, pero ahora se ha extendido a la normalización de un procedimiento de codificación más versátil para imágenes en movimiento que incluye formas de imágenes entrelazadas y velocidades binarias superiores.

En la codificación de imágenes en movimiento mencionada anteriormente, la cantidad de información se comprime aprovechando las redundancias en las direcciones espacial y temporal. Aquí, se usa la codificación de predicción entre imágenes como un procedimiento de uso de las redundancias temporales. En la codificación de predicción entre imágenes, una imagen se codifica usando una imagen en sentido inverso o en sentido directo temporalmente como imagen de referencia. Se estima el movimiento (un vector de movimiento) de la imagen actual que se codificará a partir de la imagen de referencia, y se calcula la diferencia entre la imagen obtenida por la compensación de movimiento y la imagen actual. A continuación, se eliminan las redundancias espaciales de esta diferencia, con lo que se comprime la cantidad de información de la imagen en movimiento.

El documento "H.26L Test Model Long Term Number 6 (TML-6) draft0" (Sector de Normalización de Telecomunicaciones de ITU, Ginebra, Suiza, VCEG-L45d0.doc, 3 de marzo de 2001) es una descripción de un procedimiento de codificación de referencia que se usará para el desarrollo de una recomendación-H.26L de ITU-T para un nuevo procedimiento de compresión. La configuración básica del algoritmo es similar a H.263. Algunas de las diferencias con respecto a H.263 son:

- Sólo se usa una VLC regular para codificación de símbolos.

- Se usan posiciones de 1/4 píxel para predicción de movimiento.

- Se usa una serie de diferentes tamaños de bloque para predicción de movimiento.

- La codificación residual se basa en bloques 4 x 4 y se usa una transformada de números enteros.

- Pueden usarse múltiples tramas de referencia para predicción y se considera que sustituyen cualquier uso de tramas B.

En un procedimiento de codificación de imágenes en movimiento de acuerdo con MPEG-1, MPEG-2, MPEG-4, H.263, H.26L o similares, una imagen que no se codifica como predicción entre imágenes, es decir, que se codifica intraimagen, se denomina imagen I. Aquí, una imagen significa una unidad de codificación única que incluye tanto una trama como un campo. Además, una imagen que se codifica como predicción entre imágenes con referencia a una imagen se denomina imagen P, y una imagen que se codifica como predicción entre imágenes con referencia a las dos imágenes procesadas anteriormente se denomina imagen B.

La fig. 1 es un diagrama que muestra una relación predictiva entre imágenes en el procedimiento de codificación de imágenes en movimiento mencionado anteriormente.

En la fig. 1, una línea vertical indica una imagen, con un tipo de imagen (I, P o B) indicado en la parte inferior derecha de la misma. Además, la fig. 1 indica que una imagen a la que apunta una flecha se codifica como predicción entre imágenes usando una imagen situada en el otro extremo de la punta de flecha como imagen de referencia. Por ejemplo, una imagen B que es la segunda desde la izquierda se codifica usando la primera imagen I y la cuarta imagen P como imágenes de referencia.

En el procedimiento de codificación de imágenes en movimiento de acuerdo con MPEG-4, H.26L o similar, puede seleccionarse un modo de codificación denominado modo directo para codificar una imagen B.

Se explicará un procedimiento de codificación de predicción entre imágenes en modo directo con referencia a la fig. 2.

La fig. 2 es una ilustración para explicar el procedimiento de codificación de predicción entre imágenes en modo directo.

Se supone ahora que un bloque C en una imagen B3 se codifica en modo directo. En este caso, un vector de movimiento MVP de un bloque X en una imagen de referencia (una imagen P4 que es una imagen de referencia en sentido inverso, en este caso) que ha sido codificada inmediatamente antes de que se aproveche la imagen B3, en el que el bloque X está situado junto al bloque C. El vector de movimiento MVP es un vector de movimiento que se usó cuando se codificó el bloque X, y se refiere a una imagen P1. El bloque C se predice bidireccionalmente a partir de las imágenes de referencia, es decir, la imagen P1 y la imagen P4, usando vectores de movimiento paralelos al vector de movimiento MVP. Los vectores de movimiento usados para codificar el bloque C son, en este caso, un vector de movimiento MVFc para la imagen P1 y un vector de movimiento MVBC para la imagen P4.

En el procedimiento de codificación de imágenes en movimiento de acuerdo con MPEG-4, H.26L o similar, se codifica una diferencia entre un valor predictivo obtenido de vectores de movimiento de bloques vecinos y un vector de movimiento de un bloque actual que se codificará para codificar el vector de movimiento. En la siguiente descripción, un "valor predictivo" indica un valor predictivo de un vector de movimiento. Como los vectores de movimiento de bloques vecinos tienen dirección y movimiento similares en muchos casos, la cantidad de codificación del vector de movimiento puede reducirse codificando la diferencia entre el valor predictivo obtenido y los vectores de movimiento de los bloques vecinos.

Aquí, se explicará un procedimiento de codificación de vectores de movimiento en MPEG-4 con referencia a la fig. 3.

La fig. 3 es una ilustración para explicar un procedimiento para codificar un vector de movimiento MV de un bloque actual A que se codificará en MPEG-4.

5

En las fig. 3A-3D, los bloques indicados por una línea gruesa son macrobloques de 16 x 16 píxeles, y existen 4 bloques de 8 x 8 píxeles en cada macrobloque. Aquí, se supone que un vector de movimiento se obtiene en un nivel de un bloque de 8 x 8 píxeles.

10 Según se muestra en la fig. 3A, en cuanto a un bloque actual A situado arriba a la izquierda en un macrobloque, se codifica una diferencia entre un valor predictivo y un vector de movimiento MV del bloque actual A, en la que el valor predictivo se calcula a partir de un vector de movimiento MVb de un bloque vecino B a la izquierda del bloque actual A, un vector de movimiento MVc de un bloque vecino C justo encima del bloque actual A y un vector de movimiento MVd de un bloque vecino D encima y a la derecha del bloque actual A.

15

Análogamente, según se muestra en la fig. 3B, en cuanto a un bloque actual A situado arriba a la derecha en un macrobloque, se codifica una diferencia entre un valor predictivo y un vector de movimiento MV del bloque actual A, en el que el valor predictivo se calcula a partir de un vector de movimiento MVb de un bloque vecino B a la izquierda del bloque actual A, un vector de movimiento MVc de un bloque vecino C justo encima del bloque actual A y un vector de movimiento MVd de un bloque vecino D encima y a la derecha del bloque actual A.

20

Según se muestra en la fig. 3C, en cuanto a un bloque actual A situado abajo a la izquierda en un macrobloque, se codifica una diferencia entre un valor predictivo y un vector de movimiento MV del bloque actual A, en el que el valor predictivo se calcula a partir de un vector de movimiento MVb de un bloque vecino B a la izquierda del bloque actual A, un vector de movimiento MVc de un bloque vecino C justo encima del bloque actual A y un vector de movimiento MVd de un bloque vecino D encima y a la derecha del bloque actual A.

25

Según se muestra en la fig. 3D, en cuanto a un bloque actual A situado abajo a la derecha en un macrobloque, se codifica una diferencia entre un valor predictivo y un vector de movimiento MV del bloque actual A, en el que el valor predictivo se calcula a partir de un vector de movimiento MVb de un bloque vecino B a la izquierda del bloque actual A, un vector de movimiento MVc de un bloque vecino C encima y a la izquierda del bloque actual A y un vector de movimiento MVd de un bloque vecino D justo encima del bloque actual A. Aquí, el valor predictivo se calcula usando las medianas obtenidas de las componentes horizontal y vertical de estos tres vectores de movimiento MVb, MVc y MVd respectivamente.

35

A continuación, se explicará un procedimiento de codificación de vectores de movimiento en H.26L que ha sido desarrollado para normalización con referencia a la fig. 4.

La fig. 4 es una ilustración para explicar un procedimiento para codificar un vector de movimiento MV de un bloque actual A en H.26L.

40

Un bloque actual A es un bloque de 4 x 4 píxeles, 8 x 8 píxeles o 16 x 16 píxeles, y un vector de movimiento de este bloque actual A se codifica usando un vector de movimiento de un bloque vecino B que incluye un píxel b situado a la izquierda del bloque actual A, un vector de movimiento de un bloque vecino C que incluye un píxel c situado justo encima del bloque actual A y un vector de movimiento de un bloque vecino D que incluye un píxel d situado encima y a la derecha del bloque actual A. Obsérvese que los tamaños de los bloques vecinos B, C y D no se limitan a los mostrados en la fig. 4 con líneas de puntos.

45

La fig. 5 es un organigrama que muestra el procedimiento de codificación del vector de movimiento MV del bloque actual A usando los vectores de movimiento de los bloques vecinos según se menciona anteriormente.

50

Primero, se especifica el bloque vecino que se refiere a la imagen a la que se refiere el bloque actual A entre los bloques vecinos B, C y D (Etapa S502), y se determina el número de bloques vecinos especificados (Etapa S504).

55 Cuando el número de los bloques vecinos determinados en la Etapa S504 es 1, el vector de movimiento de ese bloque vecino que se refiere a la misma imagen se considera un valor predictivo del vector de movimiento MV del bloque actual A (Etapa S506).

Cuando el número de los bloques vecinos determinado en la Etapa S505 es un valor distinto a 1, el vector de

movimiento del bloque vecino que se refiere a otra imagen distinta que la imagen a la que se refiere el bloque actual A, entre los bloques vecinos B, C y D, se considera 0 (Etapla S507). Y la mediana de los vectores de movimiento de los bloques vecinos B, C y D se considera un valor predictivo del vector de movimiento del bloque actual A (Etapla S508).

5

Usando el valor predictivo obtenido en la Etapla S506 o en la Etapla S508 de esta manera, se calcula la diferencia entre el valor predictivo y el vector de movimiento MV del bloque actual A y se codifica la diferencia (Etapla S510).

Según se describe anteriormente, en los procedimientos de codificación de vectores de movimientos de acuerdo con MPEG-4 y H.26L, los vectores de movimiento de bloques vecinos se aprovechan durante la codificación de un vector de movimiento de un bloque actual que se va a codificar.

Sin embargo, existen casos en que los vectores de movimiento de bloques vecinos no se codifican. Por ejemplo, son casos en los que un bloque vecino se codifica intraimagen, una imagen B se codifica en modo directo, y una imagen P se codifica en modo de salto. En estos casos, los bloques vecinos se codifican usando los vectores de movimiento de otros bloques excepto cuando se codifican intraimagen, es decir, los bloques vecinos se codifican usando sus propios vectores de movimiento basándose en el resultado de la estimación del movimiento.

Así, según el procedimiento tradicional de codificación de vectores de movimiento mencionado anteriormente, un vector de movimiento de un bloque actual se codifica del modo siguiente: cuando existe un bloque vecino, de entre tres bloques vecinos, que no tiene vector de movimiento basándose en el resultado anterior de estimación del movimiento y ha sido codificado usando vectores de movimiento de otros bloques, el vector de movimiento de ese bloque vecino se considera 0. Cuando existen dos de dichos bloques vecinos, el vector de movimiento del bloque vecino restante se usa como un valor predictivo. Y cuando existen tres bloques vecinos, el vector de movimiento se codifica considerando un valor predictivo de 0.

Sin embargo, en modo directo o modo de salto, la compensación de movimiento se efectúa realmente como en el caso en que se usa un vector de movimiento de un bloque vecino en sí basándose en el resultado de la estimación, aunque la información del vector de movimiento no se codifica. Como consecuencia, en el procedimiento tradicional anterior, si un bloque vecino se codifica en modo directo o modo de salto, el vector de movimiento del bloque vecino no se usa como candidato para un valor predictivo. Así, existe el problema de obtener un valor predictivo impreciso de un vector de movimiento durante la codificación del vector de movimiento, y causar así una menor eficacia de codificación.

La presente invención se concibe para resolver este problema, y el objeto de la misma es proporcionar un procedimiento de codificación de vectores de movimiento y un procedimiento de decodificación de vectores de movimiento para obtener un valor predictivo más preciso para una eficacia de codificación más alta.

Descripción de la invención

40

Con el fin de conseguir el objeto anterior, el procedimiento de codificación de vectores de movimiento según la presente invención es un procedimiento de codificación de vectores de movimiento para codificación de un vector de movimiento de un bloque actual en una imagen en movimiento, que comprende: una etapa de especificación de bloque vecino para especificar un bloque vecino que está situado en la vecindad del bloque actual y ya ha sido codificado; una etapa de valoración para valorar si el bloque vecino ha sido codificado o no usando un vector de movimiento de otro bloque; una etapa de predicción para obtener un vector de movimiento predictivo del bloque actual usando un vector de movimiento calculado a partir del vector de movimiento de dicho otro bloque como un vector de movimiento del bloque vecino, cuando se valora en la etapa de valoración que el bloque vecino ha sido codificado usando el vector de movimiento de dicho otro bloque; y una etapa de codificación para codificar el vector de movimiento del bloque actual usando el vector de movimiento predictivo.

Como resultado, cuando un vector de movimiento de un bloque actual se codifica usando un vector de movimiento predictivo obtenido de vectores de movimiento de bloques vecinos, si alguno de los bloques vecinos ha sido codificado usando vectores de movimiento de otros bloques, el vector de movimiento del bloque vecino no se considera que es 0 sino que es el vector de movimiento calculado a partir de los vectores de movimiento de los otros bloques. Por tanto, puede obtenerse un vector de movimiento predictivo más preciso, y así puede mejorarse la eficacia de codificación del vector de movimiento.

Además, el procedimiento de decodificación de vectores de movimiento según la presente invención es un

procedimiento de decodificación de vectores de movimiento para decodificar un vector de movimiento codificado de un bloque actual en una imagen en movimiento, que comprende: una etapa de especificación de bloque vecino para especificar un bloque vecino que está situado en la vecindad del bloque actual y que ya ha sido decodificado, una etapa de valoración para valorar si el bloque vecino ha sido codificado o no usando un vector de movimiento de otro bloque; una etapa de predicción para obtener un vector de movimiento predictivo del bloque actual usando un vector de movimiento calculado a partir del vector de movimiento de dicho otro bloque como un vector de movimiento del bloque vecino, cuando se valora en la etapa de valoración que el bloque vecino ha sido codificado usando el vector de movimiento de dicho otro bloque; y una etapa de decodificación para decodificar el vector de movimiento codificado del bloque actual usando el vector de movimiento predictivo.

Como resultado, el vector de movimiento que ha sido codificado según el procedimiento de codificación de vectores de movimiento de la presente invención puede decodificarse adecuadamente, y así el valor práctico del mismo es elevado.

Obsérvese que la presente invención también puede realizarse como un aparato de codificación de imágenes en movimiento y un programa que usan el procedimiento de codificación de vectores de movimiento mencionado anteriormente, y un medio de almacenamiento que almacena el programa, y un aparato de decodificación de imágenes en movimiento y un programa que usan el procedimiento de decodificación de vectores de movimiento mencionado anteriormente, y un medio de almacenamiento que almacena el programa.

Breve descripción de los dibujos

La fig. 1 es un diagrama que muestra una relación predictiva entre imágenes en un procedimiento de codificación de imágenes en movimiento;

la fig. 2 es una ilustración para explicar el procedimiento de predicción entre imágenes en modo directo;

la fig. 3 es una ilustración para explicar un procedimiento para codificar un vector de movimiento de un bloque actual en MPEG-4;

la fig. 4 es una ilustración para explicar un procedimiento para codificar un vector de movimiento de un bloque actual en H.26L;

la fig. 5 es un organigrama que muestra el procedimiento de codificación en H26L;

la fig. 6 es un diagrama de bloques que muestra una estructura de un aparato de codificación de imágenes en movimiento en una primera forma de realización de la presente invención;

la fig. 7 es un diagrama que muestra cómo se introducen y se producen las imágenes en una memoria de trama en la primera forma de realización;

la fig. 8 es un organigrama que muestra el funcionamiento de una unidad de codificación de vectores de movimiento en la primera forma de realización;

la fig. 9 es una ilustración para explicar cómo codificar un bloque vecino en modo de salto en la primera forma de realización;

la fig. 10 es una ilustración para explicar la codificación de predicción entre imágenes usando vectores de movimiento bidireccionales en la primera forma de realización;

la fig. 11 es una ilustración para explicar cómo codificar un bloque vecino en modo directo temporal en la primera forma de realización;

la fig. 12 es una ilustración para explicar cómo codificar un bloque vecino en modo directo espacial en la primera forma de realización;

la fig. 13 es un organigrama que muestra otro funcionamiento de la unidad de codificación de vectores de movimiento en la primera forma de realización;

la fig. 14 es un diagrama de bloques que muestra una estructura de un aparato de decodificación de imágenes en movimiento en una segunda forma de realización de la presente invención;

la fig. 15 es un organigrama que muestra un funcionamiento de una unidad de decodificación de vectores de movimiento en la segunda forma de realización;

la fig. 16 es una ilustración para explicar cómo se introducen y se producen las imágenes a partir del aparato de decodificación de imágenes en movimiento en la segunda forma de realización;

10 la fig. 17 es un organigrama que muestra otro funcionamiento de la unidad de decodificación de vectores de movimiento en la segunda forma de realización;

la fig. 18 es una ilustración de un medio de grabación en una tercera forma de realización de la presente invención;

15 la fig. 19 es un diagrama de bloques que muestra una configuración general de un sistema de suministro de contenidos en una cuarta forma de realización de la presente invención;

la fig. 20 es una vista frontal de un teléfono móvil en la cuarta forma de realización;

20 la fig. 21 es un diagrama de bloques del teléfono móvil en la cuarta forma de realización;

la fig. 22 es un diagrama de bloques que muestra una configuración general de un sistema de difusión digital en la cuarta forma de realización.

25 **Mejor modo de realizar la invención**

(Primera forma de realización)

Se explicará un aparato de codificación de imágenes en movimiento en una primera forma de realización de la presente invención con referencia a las figuras.

La fig. 6 es un diagrama de bloques del aparato de codificación de imágenes en movimiento en la primera forma de realización de la presente invención.

35 Este aparato de codificación de imágenes en movimiento 100 pretende mejorar la eficacia de codificación mejorando la precisión de un valor predictivo de un vector de movimiento, e incluye una memoria de trama 101, una unidad de cálculo de diferencias 102, una unidad de codificación de errores de predicción 103, una unidad de generación de flujos de bits 104, una unidad de decodificación de errores de predicción 105, una unidad de suma 106, una memoria de trama 107, una unidad de estimación de vectores de movimiento 108, una unidad de selección de modo
40 109, una unidad de control de codificación 110, conmutadores 111~115, una unidad de almacenamiento de vectores de movimiento 116 y una unidad de codificación de vectores de movimiento 117.

La memoria de trama 101 es una memoria de imágenes para guardar imágenes introducidas sobre una base de imagen por imagen, y reordena las imágenes introducidas y obtenidas en orden de tiempo en orden de codificación
45 para la salida. Las imágenes se reordenan bajo el control de la unidad de control de codificación 110.

La fig. 7A muestra cómo se introducen las imágenes en la memoria de trama 101.

En la fig. 7A, las líneas verticales muestran imágenes, y un alfabeto y un número en la parte inferior derecha de cada
50 imagen indica un tipo de imagen (I, P o B) y un número de imagen en orden de tiempo. Las imágenes introducidas en la memoria de trama 101 se reordenan en orden de codificación. Las imágenes se reordenan en orden de codificación basándose en las relaciones de referencia en codificación de predicción entre imágenes, es decir, las imágenes se reordenan de manera que las imágenes usadas como imágenes de referencia se codifican antes que las imágenes que se refieren a dichas imágenes de referencia. Por ejemplo, las relaciones de referencia de las
55 imágenes P7~P13 se muestran mediante flechas en la fig. 7A. En la fig. 7A, las puntas de flecha indican las imágenes que se refieren a imágenes de referencia, y los otros extremos de las flechas indican las imágenes de referencia. En este caso, las imágenes mostradas en la fig. 7A se reordenan en las mostradas en la fig. 7B.

La fig. 7B muestra las imágenes introducidas según se muestra en la fig. 7A y reordenadas. Las imágenes

reordenadas en la memoria de trama 101 se leen según una base de macrobloque. En este caso, un macrobloque tiene un tamaño de 16 píxeles horizontales x 16 verticales.

5 La unidad de cálculo de diferencias 102 obtiene datos de formas de imágenes de todos los macrobloques de la memoria de trama 101 por medio del conmutador 111, y también obtiene una forma de imagen de compensación de movimiento de la unidad de selección de modo 109. A continuación, la unidad de cálculo de diferencias 102 calcula la diferencia entre los datos de formas de imágenes y la forma de imagen de compensación de movimiento según una base de macrobloque para generar una forma de imagen de error de predicción para salida.

10 La unidad de codificación de errores de predicción 103 realiza procesamiento de codificación que incluye transformación de frecuencias como transformación de cosenos discreta y cuantificación en los datos de formas de imágenes obtenidos de la memoria de trama 101 por medio del conmutador 112 y la forma de imagen de error de predicción obtenida por la unidad de cálculo de diferencias 102, de manera que se creen datos codificados. Por ejemplo, la transformación de frecuencias y la cuantificación se realizan en una unidad de 8 píxeles horizontales x 8
15 verticales. A continuación, la unidad de codificación de errores de predicción 103 produce los datos codificados para la unidad de generación de flujos de bits 104 y la unidad de decodificación de errores de predicción 105.

La unidad de generación de flujos de bits 104 realiza codificación de longitud variable en los datos codificados producidos desde la unidad de codificación de errores de predicción 103, convierte los datos de la misma en un
20 formato de flujo de bits para salida, y añade además información sobre vectores de movimiento introducidos desde la unidad de codificación de vectores de movimiento 117, información sobre un modo de codificación introducido desde la unidad de selección de modo 109, información de encabezamiento y otros, con el fin de generar un flujo de bits.

25 La unidad de decodificación de errores de predicción 105 cuantifica inversamente los datos codificados producidos desde la unidad de codificación de errores de predicción 103, y a continuación realiza transformación de frecuencias inversa como transformación de cosenos discreta inversa de manera que se decodifican en una forma de imagen de error de predicción.

30 La unidad de suma 106 añade la forma de imagen de compensación de movimiento a la forma de imagen de error de predicción obtenida como resultado de decodificación, y produce una imagen decodificada que es datos de formas de imágenes que indican una forma de una imagen que ha sido codificada y decodificada.

La memoria de trama 107 es una memoria de imágenes que guarda, según una base de imagen por imagen,
35 imágenes usadas como imágenes de referencia cuando se codifican otras imágenes, entre las imágenes decodificadas producidas desde la unidad de suma 106.

La unidad de estimación de vectores de movimiento 108 estima vectores de movimiento de cada bloque en un macrobloque actual que se codificará, usando las imágenes decodificadas acumuladas en la memoria de trama 107
40 como imágenes de referencia. Los vectores de movimiento estimados se producen para la unidad de selección de modo 109.

La unidad de selección de modo 109 determina un modo de codificación del macrobloque usando los vectores de movimiento estimados por la unidad de estimación de vectores de movimiento 108. Aquí, el modo de codificación
45 significa un procedimiento para codificar un macrobloque. Por ejemplo, cuando una imagen actual es una imagen P, la unidad de selección de modo 109 determina un modo de codificación de entre los siguientes: codificación intraimagen, codificación de predicción entre imágenes usando vectores de movimiento y modo de salto (codificación de predicción entre imágenes en el que no se codifica ningún vector de movimiento de un bloque actual dado que la codificación de predicción se realiza usando un vector de movimiento obtenido de vectores de
50 movimiento de otros bloques, y no se codifica ningún valor de coeficiente dado que todos los valores de coeficientes son 0 como resultado de la codificación de error de predicción). En general, se determina un modo de codificación de manera que se reduzca al mínimo un error de codificación usando una cantidad de bits predeterminada.

La unidad de selección de modo 109 produce el modo de codificación determinado para la unidad de generación de flujos de bits 104, y produce los vectores de movimiento usados para ese modo de codificación para la unidad de
55 codificación de vectores de movimiento 117, respectivamente. Cuando el modo de codificación determinado es una codificación de predicción entre imágenes que usa vectores de movimiento, la unidad de selección de modo 109 almacena además los vectores de movimiento y el modo de codificación usado para esa codificación de predicción entre imágenes en la unidad de almacenamiento de vectores de movimiento 116.

Además, la unidad de selección de modo 109 realiza compensación de movimiento basándose en el modo de codificación determinado y en los vectores de movimiento estimados por la unidad de estimación de vectores de movimiento 108 de manera que se crea una forma de imagen de compensación de movimiento, y se produce la forma de imagen de compensación de movimiento para la unidad de cálculo de diferencias 102 y la unidad de suma 106. Sin embargo, si se selecciona codificación de imágenes, no se produce ninguna forma de imagen de compensación de movimiento. Cuando se selecciona codificación intraimagen, la unidad de selección de modo 109 controla además el conmutador 111 y el conmutador 112 para conectarse a un terminal "a" y a un terminal "c" respectivamente, y cuando se selecciona codificación de predicción entre imágenes, controla el conmutador 111 y el conmutador 112 para conectarse a un terminal "b" y un terminal "d" respectivamente. La compensación de movimiento mencionada anteriormente se realiza según una base de bloque por bloque (8 x 8 píxeles en este caso).

La unidad de control de codificación 110 determina un tipo de imagen (I, P o B) usado para codificación de una imagen introducida, y controla los conmutadores 113, 114 y 115 dependiendo del tipo de imagen. Aquí, un tipo de imagen se determina en general usando un procedimiento para asignar un tipo de imagen periódicamente, por ejemplo.

La unidad de almacenamiento de vectores de movimiento 116 obtiene los vectores de movimiento usados para codificación de predicción entre imágenes y el modo de codificación de la unidad de selección de modo 109, y los almacena.

Cuando la unidad de selección de modo 109 selecciona codificación de predicción entre imágenes usando vectores de movimiento, la unidad de codificación de vectores de movimiento 117 codifica un vector de movimiento de un bloque actual mediante los procedimientos descritos en la fig. 3 y la fig. 4. Según se describe anteriormente, la unidad de codificación de vectores de movimiento 117 especifica tres bloques vecinos del bloque actual, determina un valor predictivo basándose en los vectores de movimiento de los bloques vecinos, y codifica una diferencia entre el valor predictivo y el vector de movimiento del bloque actual que se codificará.

Cuando se codifica un vector de movimiento de un bloque actual, si un bloque vecino se codifica usando vectores de movimiento de otros bloques, como modo de salto y modo directo, la unidad de codificación de vectores de movimiento 117 en la presente forma de realización no considera que el vector de movimiento del bloque vecino es 0 como hace la técnica convencional, sino que trata un vector de movimiento obtenido de los vectores de movimiento de los otros bloques como el vector de movimiento del bloque vecino cuando lo codifica.

La fig. 8 es un organigrama que muestra el funcionamiento general de la unidad de codificación de vectores de movimiento 117 en la presente forma de realización.

En primer lugar, la unidad de codificación de vectores de movimiento 117 especifica tres bloques vecinos codificados previamente de un bloque actual (Etapa S100).

La unidad de codificación de vectores de movimiento 117 valora si cada uno de los bloques vecinos especificados es un bloque vecino Ba que ha sido codificado usando vectores de movimiento de otros bloques o un bloque vecino Bb que ha sido codificado sin usar vectores de movimiento de otros bloques (Etapa S102).

Como resultado, la unidad de codificación de vectores de movimiento 117 determina si los tres bloques vecinos especificados incluyen o no un bloque vecino Ba (Etapa S104).

Cuando se valora en la Etapa S104 que se incluye el bloque vecino Ba (Y en la Etapa S104), la unidad de codificación de vectores de movimiento 117 obtiene un valor predictivo de los vectores de movimiento de los tres bloques vecinos tratando un vector de movimiento obtenido de los vectores de movimiento de los otros bloques como un vector de movimiento del bloque vecino Ba para codificarlo, según se menciona anteriormente (Etapa S106).

Por otra parte, cuando se valora en la Etapa S104 que no se incluye el bloque vecino Ba (N en la Etapa S104), la unidad de codificación de vectores de movimiento 117 obtiene un valor predictivo de vectores de movimiento obtenido basándose en la estimación de movimiento de tres bloques vecinos respectivos Bb y la selección de modo (Etapa S108).

A continuación, la unidad de codificación de vectores de movimiento 117 codifica una diferencia entre el vector de

movimiento del bloque actual y el valor predictivo obtenido en las Etapas S106 o S108 (Etapa S110). La unidad de codificación de vectores de movimiento 117 también produce el vector de movimiento codificado como anteriormente para la unidad de generación de flujos de bits 104.

- 5 Aquí, el procesamiento de codificación mencionado anteriormente por el aparato de codificación de imágenes en movimiento 100 se explicará específicamente tomando la codificación de una imagen P13 y una imagen B11 según se muestra en la fig. 7 como ejemplo.

(Codificación de la imagen P13)

- 10 Como la imagen P13 es una imagen P, el aparato de codificación de imágenes en movimiento 100 codifica la imagen 13 por codificación de predicción entre imágenes usando otra imagen como imagen de referencia. En este caso, la imagen de referencia es una imagen P10. Esta imagen P10 ya ha sido codificada, y la imagen decodificada de la misma se almacena en la memoria de trama 107.

- 15 Cuando se codifica una imagen P, la unidad de control de codificación 110 controla los conmutadores 113, 114 y 115 para que estén ACTIVOS. Por tanto, los macrobloques de la imagen P13 que se leen desde la memoria de trama 101 son obtenidos por la unidad de estimación de vectores de movimiento 108, la unidad de selección de modo 109 y la unidad de cálculo de diferencias 102.

- 20 La unidad de estimación de vectores de movimiento 108 estima el vector de movimiento de cada bloque en el macrobloque usando la imagen decodificada de la imagen P10 almacenada en la memoria de trama 107 como una imagen de referencia, y produce el vector de movimiento estimado para la unidad de selección de modo 109.

- 25 La unidad de selección de modo 109 determina un modo de codificación del macrobloque en la imagen P13 usando el vector de movimiento estimado por la unidad de estimación de vectores de movimiento 108. Como la imagen P13 es una imagen P, la unidad de selección de modo 109 determina, según se menciona anteriormente, un modo de codificación entre los siguientes: codificación intraimagen, codificación de predicción entre imágenes usando vectores de movimiento y modo de salto (una codificación de predicción entre imágenes en la que no se codifica ningún vector de movimiento de un bloque actual dado que la codificación de predicción se realiza usando un vector de movimiento obtenido de los vectores de movimiento de otros bloques, y no se codifica ningún valor de coeficiente porque todos los valores de coeficientes son 0 como resultado de la codificación de error de predicción).

- 35 Cuando la unidad de selección de modo 109 selecciona codificación de predicción entre imágenes usando vectores de movimiento, la unidad de codificación de vectores de movimiento 117 en la presente forma de realización codifica el vector de movimiento del bloque actual en la imagen P13 por el procedimiento según se explica en la fig. 3. Cuando un bloque vecino del bloque actual se codifica en modo de salto, la unidad de codificación de vectores de movimiento 117 no considera que el vector de movimiento del bloque vecino es 0, sino que trata un vector de movimiento obtenido de otros bloques para codificar el bloque vecino como un vector de movimiento de ese bloque.

- 40 Se explicará un procedimiento de codificación de un vector de movimiento de un bloque actual usado cuando un bloque vecino se codifica en modo de salto.

La fig. 9 es una ilustración para explicar cómo codificar un bloque vecino C en modo de salto.

- 45 Según se muestra en la fig. 9, cuando un bloque vecino C en la imagen P13 se codifica en modo de salto, se calcula una mediana de un vector de movimiento MV_e de un bloque E, un vector de movimiento MV_f de un bloque F y un vector de movimiento MV_g de un bloque G, que están situados en la vecindad del bloque vecino C, y el bloque vecino C se codifica usando un vector de movimiento MV_{cm} que indica la mediana. Aquí, se obtiene una mediana de vectores de movimiento calculando medianas de componentes horizontales y verticales de los vectores de movimiento respectivamente, por ejemplo.

- 50 Cuando se codifica el vector de movimiento del bloque actual A según se muestra en la fig. 9, la unidad de codificación de vectores de movimiento 117 especifica los tres bloques vecinos B, C y D del bloque actual A (como para las posiciones de los bloques B, C y D, véanse fig. 3 y fig. 4), y valora si cada uno de los bloques vecinos B, C y D es o no un bloque que ha sido codificado usando vectores de movimiento de otros bloques. Como resultado, cuando se valora que sólo el bloque vecino C se codifica en modo de salto, es decir, se codifica usando otros bloques, la unidad de codificación de vectores de movimiento 117 trata la mediana (un vector de movimiento MV_{cm}) calculada a partir de los vectores de movimiento de los otros bloques E, F y G para codificar el bloque vecino C

como un vector de movimiento del bloque vecino C, según se menciona anteriormente, y calcula la mediana del vector de movimiento MVcm y los vectores de movimiento de los bloques vecinos B y D para considerarlos como un valor predictivo del vector de movimiento del bloque actual A. A continuación, la unidad de codificación de vectores de movimiento 117 codifica una diferencia entre el valor predictivo y el vector de movimiento del bloque actual A.

5

La unidad de almacenamiento de vectores de movimiento 116 almacena modos de codificación de bloques codificados. La unidad de codificación de vectores de movimiento 117 valora si cada uno de los bloques vecinos B, C y D es un bloque codificado usando o no vectores de movimiento de otros bloques basándose en los modos de codificación almacenados en la unidad de almacenamiento de vectores de movimiento 116. La unidad de almacenamiento de vectores de movimiento 116 almacena además vectores de movimiento de bloques que han sido codificados sin usar vectores de movimiento de otros bloques pero usando sus propios vectores de movimiento estimados a partir de imágenes de referencia. Para ser más concreto, la unidad de almacenamiento de vectores de movimiento 116 almacena los vectores de movimiento MVe, MVf y MVg de los bloques E, F y G, y la unidad de codificación de vectores de movimiento 117 calcula el vector de movimiento MVcm mencionado anteriormente del bloque vecino C usando estos vectores de movimiento almacenados en la unidad de almacenamiento de vectores de movimiento 116 cuando se codifica el vector de movimiento del bloque actual A. Obsérvese que como para una imagen que ha sido codificada usando vectores de movimiento de otros bloques, un vector de movimiento de la misma que se obtiene calculando una mediana de los vectores de movimiento de los otros bloques puede almacenarse en la unidad de almacenamiento de vectores de movimiento 116 con antelación. En este caso, como la unidad de almacenamiento de vectores de movimiento 116 almacena el vector de movimiento MVcm con antelación, la unidad de codificación de vectores de movimiento 117 no necesita calcular el vector de movimiento MVcm del bloque vecino C pero puede usar el vector de movimiento MVcm almacenado en la unidad de almacenamiento de vectores de movimiento 116 directamente como un vector de movimiento del bloque vecino C, cuando se codifica el vector de movimiento del bloque actual A.

25

Por otra parte, una forma de imagen de error de predicción que indica una diferencia entre un macrobloque actual en la imagen P13 y una forma de imagen de compensación de movimiento se codifican mediante la unidad de codificación de errores de predicción 103 y se generan como datos codificados, y se añade la información sobre el vector de movimiento codificada según se menciona anteriormente a los datos codificados por la unidad de generación de flujos de bits 104. Sin embargo, no se añaden a los datos codificados una diferencia entre un macrobloque que ha sido codificado en modo de salto y una forma de imagen de compensación de movimiento es 0, e información sobre el vector de movimiento.

Los restantes macrobloques de la imagen P13 se codifican de la misma manera. Después de completar la codificación de todos los macrobloques de la imagen P13, sigue la codificación de la imagen B11.

(Codificación de la imagen B11)

Como la imagen B11 es una imagen B, el aparato de codificación de imágenes en movimiento 100 codifica la imagen B11 mediante codificación de la predicción entre imágenes usando otras dos imágenes como imágenes de referencia. En este caso, las imágenes de referencia son la imagen P10 situada delante de la imagen B11 y la imagen P13 situada detrás de la imagen B11. Estas imágenes P10 y P13 ya han sido codificadas, y las imágenes decodificadas de las mismas se almacenan en la memoria de trama 107.

Cuando se codifica una imagen B, la unidad de control de codificación 110 controla que el conmutador 113 esté ACTIVO y que los conmutadores 114 y 115 estén INACTIVOS. Por tanto, los macrobloques de la imagen B11 que se leen desde la memoria de trama 101 son obtenidos por la unidad de estimación de vectores de movimiento 108, la unidad de selección de modo 109 y la unidad de cálculo de diferencias 102.

La unidad de estimación de vectores de movimiento 108 estima el vector de movimiento en sentido inverso y el vector de movimiento en sentido directo de cada bloque en un macrobloque usando una imagen decodificada de la imagen P10 almacenada en la memoria de trama 107 como una imagen de referencia en sentido inverso y una imagen decodificada de la imagen P13 como una imagen de referencia en sentido directo, y produce los vectores de movimiento estimados en sentido inverso y en sentido directo para la unidad de selección de modo 109.

55

La unidad de selección de modo 109 determina un modo de codificación del macrobloque de la imagen B11 usando los vectores de movimiento en sentido inverso y en sentido directo estimados por la unidad de estimación de vectores de movimiento 108. Como la imagen B11 es una imagen B, la unidad de selección de modo 109 determina un modo de codificación entre los siguientes: codificación intraimagen, codificación de predicción entre imágenes

usando vectores de movimiento anteriores, codificación de predicción entre imágenes usando vectores de movimiento posteriores, codificación de predicción entre imágenes usando vectores de movimiento bidireccionales y modo directo (codificación de predicción entre imágenes en la que la compensación de movimiento se realiza usando un vector de movimiento obtenido de los vectores de movimiento de otros bloques y no se codifica ningún vector de movimiento), por ejemplo.

Cuando la unidad de selección de modo 109 selecciona codificación de predicción entre imágenes usando vectores de movimiento, la unidad de codificación de vectores de movimiento 117 en la presente forma de realización codifica los vectores de movimiento del bloque actual en la imagen B11 por el procedimiento que se explica en la fig. 3.

Más específicamente, cuando la unidad de selección de modo 109 selecciona codificación de predicción entre imágenes usando vectores de movimiento bidireccionales, la unidad de codificación de vectores de movimiento 117 codifica los vectores de movimiento del bloque actual de la siguiente manera.

La fig. 10 es una ilustración para explicar la codificación de predicción entre imágenes usando vectores de movimiento bidireccionales.

Cuando se codifican vectores de movimiento de un bloque actual A, la unidad de codificación de vectores de movimiento 117 codifica un vector de movimiento en sentido inverso MVF y un vector de movimiento en sentido directo MVB.

Para ser más concreto, la unidad de codificación de vectores de movimiento 117 considera una mediana de vectores de movimiento anteriores MVF1, MVF2 y MVF3 de los bloques vecinos B, C y D como un valor predictivo del vector de movimiento en sentido inverso MVF, y codifica una diferencia entre el vector de movimiento en sentido inverso MVF y el valor predictivo del mismo. La unidad de codificación de vectores de movimiento 117 considera también una mediana de vectores de movimiento posteriores MVB1, MVB2 y MVB3 de los bloques vecinos B, C y D como un valor predictivo del vector de movimiento en sentido directo MVB, y codifica una diferencia entre el vector de movimiento en sentido directo MVB y el valor predictivo del mismo. Aquí, la mediana de los vectores de movimiento se obtiene calculando medianas de componentes horizontales y verticales de los vectores de movimiento respectivamente, por ejemplo.

Cuando se codifican vectores de movimiento de un bloque actual en una imagen B, si un bloque vecino ha sido codificado en modo directo, la unidad de codificación de vectores de movimiento 117 en la presente forma de realización no considera que los vectores de movimiento del bloque vecino son 0, sino que considera vectores de movimiento obtenidos de otros bloques como vectores de movimiento del bloque vecino. Existen dos tipos de modo directos: modo directo temporal y modo directo espacial.

En primer lugar, se explicará cómo codificar vectores de movimiento de un bloque actual cuando un bloque vecino se codifica en modo directo temporal.

La fig. 11 es una ilustración para explicar cómo codificar el bloque vecino en modo directo temporal.

Según se muestra en la fig. 11, cuando el bloque vecino C en la imagen B11 se codifica en modo directo, se usa un vector de movimiento MVp de un bloque X, que se sitúa al lado del bloque vecino C, en la imagen P13 que es una imagen de referencia en sentido inverso recién codificada. El vector de movimiento MVp es un vector de movimiento usado para codificar el bloque X, y se almacena en la unidad de almacenamiento de vectores de movimiento 116. Este vector de movimiento MVp se refiere a la imagen P10. El bloque vecino C se codifica mediante predicción bidireccional a partir de las imágenes de referencia, la imagen P10 y la imagen P13, usando vectores de movimiento paralelos al vector de movimiento MVp. En este caso, los vectores de movimiento usados para codificar el bloque vecino C son un vector de movimiento MVFc para la imagen P10 y un vector de movimiento MVBC para la imagen P13.

En este caso cuando el vector de movimiento en sentido directo MVFc es mvf, el vector de movimiento en sentido inverso MVBC es mvb, el vector de movimiento MVp es mvp, la distancia temporal entre la imagen de referencia en sentido inverso (imagen P13) para la imagen actual (imagen B11) y la imagen de referencia (imagen P10) a la que apunta el bloque en la imagen de referencia en sentido inverso es TRD, y la distancia temporal entre la imagen actual (imagen B11) y la imagen de referencia (imagen P10) a la que apunta el bloque en la imagen de referencia en sentido inverso es TRB, mvf y mvb se calculan respectivamente mediante la Ecuación 1 y la Ecuación 2.

$$mvf = mvp \times TRB/TRD$$

Ecuación 1

$$mvp = (TRB - TRD) \times xmvp/TRD$$

Ecuación 2

5 en las que mvf y mvp representan respectivamente las componentes horizontales y las componentes verticales de los vectores de movimiento. Y los valores más indican la dirección del vector de movimiento Mvp , y los valores menos indican la dirección opuesta a la del vector de movimiento Mvp .

10 El bloque vecino C se codifica usando los vectores de movimiento MVF_c y MVB_c obtenidos según se menciona anteriormente.

15 Cuando se codifican los vectores de movimiento MVF y MVB del bloque actual A según se muestra en la fig. 10, la unidad de codificación de vectores de movimiento 117 especifica los tres bloques vecinos B, C y D del bloque actual A, y valora si cada uno de los bloques vecinos B, C y D es o no un bloque que ha sido codificado usando un vector de movimiento de otro bloque. Como resultado, cuando se valora que sólo el bloque vecino C se codifica en modo directo temporal, es decir, se codifica usando el vector de movimiento del otro bloque, la unidad de codificación de vectores de movimiento 117 trata los vectores de movimiento MVF_c y MVB_c calculados a partir del vector de movimiento Mvp del bloque X que es el otro bloque para codificar el bloque vecino C como vectores de movimiento del bloque vecino C, y calcula las medianas de los vectores de movimiento MVF_c y MVB_c y los vectores de movimiento de los bloques vecinos B y D para obtener valores predictivos de los vectores de movimiento del bloque actual A. Se obtienen por separado un valor predictivo en sentido directo y un valor predictivo en sentido inverso. A continuación, la unidad de codificación de vectores de movimiento 117 codifica diferencias entre los valores predictivos y los vectores de movimiento MVF y MVB del bloque actual A, respectivamente.

25 La unidad de almacenamiento de vectores de movimiento 116 almacena modos de codificación de bloques codificados, y basándose en los modos de codificación almacenados en esta unidad de almacenamiento de vectores de movimiento 116, la unidad de codificación de vectores de movimiento 117 valora si cada uno de los bloques vecinos B, C y D ha sido codificado o no usando vectores de movimiento de otros bloques. La unidad de almacenamiento de vectores de movimiento 116 almacena además vectores de movimiento de bloques que han sido codificados sin usar vectores de movimiento de otros bloques sino usando sus propios vectores de movimiento estimados a partir de imágenes de referencia. En otras palabras, cuando se codifican los vectores de movimiento del bloque actual A, la unidad de codificación de vectores de movimiento 117 usa los vectores de movimiento almacenados en la unidad de almacenamiento de vectores de movimiento 116 cuando son para los bloques vecinos B y D, pero si son para el bloque vecino C, lee el vector de movimiento Mvp del bloque X almacenado en la unidad de almacenamiento de vectores de movimiento 116 para calcular los vectores de movimiento MVF_c y MVB_c . Obsérvese que la unidad de almacenamiento de vectores de movimiento 116 puede almacenar con antelación vectores de movimiento calculados a partir de vectores de movimiento de otros bloques con el fin de codificar un bloque que ha sido codificado usando los vectores de movimiento de los otros bloques. En este caso, la unidad de almacenamiento de vectores de movimiento 116 almacena con antelación los vectores de movimiento MVF_c y MVB_c . Por tanto, cuando se codifican los vectores de movimiento del bloque actual A, la unidad de codificación de vectores de movimiento 117 no necesita leer el vector de movimiento Mvp del bloque X de manera que se calculen los vectores de movimiento MVF_c y MVB_c del bloque vecino C usando la Ecuación 1 y la Ecuación 2, sino que puede usar los vectores de movimiento MVF_c y MVB_c almacenados en la unidad de almacenamiento de vectores de movimiento 116 directamente como los vectores de movimiento del bloque vecino C.

45 A continuación, se explicará un procedimiento para codificar vectores de movimiento de un bloque actual en un caso en el que un bloque vecino se codifica en modo directo espacial.

50 La fig. 12 es una ilustración para explicar cómo codificar un bloque vecino en modo directo espacial.

Según se muestra en la fig. 12, cuando un bloque vecino C de la imagen B11 se codifica en modo directo espacial, se codifica usando vectores de movimiento MVF_c y MVB_c calculados basándose en medianas en las direcciones directa e inversa respectivamente que se obtienen de los vectores de movimiento MVF_e y MVB_e del bloque E, los vectores de movimiento MVF_f y MVB_f del bloque F y los vectores de movimiento MVF_g y MVB_g del bloque G, en los que los bloques E, F y G están situados en la vecindad del bloque vecino C.

Cuando se codifican los vectores de movimiento MVF y MVB del bloque actual A según se muestra en la fig. 10, la unidad de codificación de vectores de movimiento 117 especifica los tres bloques vecinos B, C y D en la vecindad del bloque actual A, y valora si cada uno de los bloques vecinos B, C y D es un bloque que ha sido codificado o no

usando vectores de movimiento de otros bloques. Como resultado, cuando la unidad de codificación de vectores de movimiento 117 valora que sólo el bloque vecino C ha sido codificado en modo directo espacial, es decir, usando vectores de movimiento de otros bloques, trata los vectores de movimiento MVFc y MVBc calculados a partir de los bloques E, F y G que son los otros bloques usados para codificar el bloque vecino C como los vectores de movimiento del bloque vecino C, calcula las medianas de los vectores de movimiento MVFc y MVBc y los vectores de movimiento de los bloques vecinos B y D, y así obtiene los valores predictivos de los vectores de movimiento del bloque actual A, según se muestra en la fig. 12. A continuación, la unidad de codificación de vectores de movimiento 117 codifica diferencias entre los valores predictivos y los vectores de movimiento MVF y MVB del bloque actual A.

- 10 La unidad de almacenamiento de vectores de movimiento 116 almacena vectores de movimiento de bloques que han sido codificados sin usar vectores de movimiento de otros bloques sino usando sus propios vectores de movimiento estimados a partir de imágenes de referencia. En otras palabras, almacena dos vectores de movimiento en las direcciones directa e inversa para cada uno de los bloques E, F y G. Cuando se codifican los vectores de movimiento del bloque actual A, la unidad de codificación de vectores de movimiento 117 calcula los vectores de movimiento MVFc y MVBc del bloque vecino C usando estos vectores de movimiento almacenados en la unidad de almacenamiento de vectores de movimiento 116. Obsérvese que la unidad de almacenamiento de vectores de movimiento 116 puede almacenar con antelación dos vectores de movimiento en las direcciones directa e inversa que se calculan basándose en medianas obtenidas a partir de vectores de movimiento de otros bloques con el fin de codificar un bloque que ha sido codificado usando los vectores de movimiento de los otros bloques. En este caso, la unidad de almacenamiento de vectores de movimiento 116 almacena con antelación los vectores de movimiento MVFc y MVBc. Por tanto, cuando se codifican los vectores de movimiento del bloque actual A, la unidad de codificación de vectores de movimiento 117 no necesita calcular los vectores de movimiento MVFc y MVBc del bloque vecino C, sino que puede usar los vectores de movimiento MVFc y MVBc almacenados en la unidad de almacenamiento de vectores de movimiento 116 directamente como los vectores de movimiento del bloque vecino C.

- Según se describe anteriormente, cuando el bloque vecino C se codifica en el modo directo temporal anterior, los vectores de movimiento de la imagen de referencia en sentido inverso (la imagen P13 en el caso anterior) de la imagen actual necesita almacenarse en la unidad de almacenamiento de vectores de movimiento 116, pero cuando el bloque vecino C se codifica en modo directo espacial, puede omitirse el almacenamiento en sí.

- Aquí, cuando se codifican los vectores de movimiento de un bloque actual, el aparato de codificación de imágenes en movimiento 100 realiza un procesamiento excepcional si un bloque vecino del bloque actual no se codifica como predicción entre imágenes, según se menciona anteriormente, sino que se codifica intraimagen.

- 35 Por ejemplo, cuando existe un bloque que ha sido codificado intraimagen en los tres bloques vecinos, la unidad de codificación de vectores de movimiento 117 del aparato de codificación de imágenes en movimiento 100 realiza un procesamiento considerando que los vectores de movimiento del bloque son 0. Cuando existen dos bloques vecinos que han sido codificados intraimagen, la unidad de codificación de vectores de movimiento 117 usa los vectores de movimiento del bloque vecino restante como valores predictivos de vectores de movimiento de un bloque actual. Además, cuando los tres bloques vecinos han sido codificados intraimagen, la unidad de codificación de vectores de movimiento 117 realiza procesamiento de codificación de los vectores de movimiento del bloque actual considerando que los valores predictivos de los mismos son 0.

- 45 Por otra parte, la forma de imagen de error de predicción que indica una diferencia entre un macrobloque actual en la imagen B11 y la forma de imagen de compensación de movimiento han sido codificadas por la unidad de codificación de errores de predicción 103 y generadas como datos codificados, y la información sobre los vectores de movimiento que ha sido codificada según se menciona anteriormente se añade a los datos codificados por la unidad de generación de flujos de bits 104. Sin embargo, la información sobre vectores de movimiento de un macrobloque que ha sido codificado en modo directo no se añade a los datos codificados.

- El procesamiento de codificación de los restantes macrobloques en la imagen B11 se realiza de la misma manera. Después de que se completa el procesamiento para todos los macrobloques de la imagen B11, sigue el procesamiento de codificación de la imagen B12.

- 55 Según se describe anteriormente, según el procedimiento de codificación de vectores de movimiento de la presente invención, un vector de movimiento de cada bloque actual se codifica usando un valor predictivo obtenido de los vectores de movimiento de los bloques vecinos codificados previamente y el vector de movimiento del bloque actual. Si alguno de los bloques vecinos ha sido codificado usando un vector de movimiento calculado a partir de vectores

de movimiento de otros bloques, por ejemplo, en modo de salto o modo directo, un valor predictivo se obtiene usando, como un vector de movimiento del bloque vecino, el vector de movimiento calculado a partir de los vectores de movimiento de los otros bloques para codificar ese bloque vecino.

- 5 En consecuencia, cuando un vector de movimiento de un bloque actual se codifica usando un valor predictivo obtenido de un vector de movimiento de un bloque vecino, si el bloque vecino se codifica usando vectores de movimiento de otros bloques, el vector de movimiento del bloque vecino no se considera 0 como en la técnica convencional, sino que se usa el vector de movimiento calculado a partir de los vectores de movimiento de los otros bloques como vector de movimiento del bloque vecino. Como resultado, se mejora la precisión del valor predictivo
10 anterior, y así puede mejorarse la eficacia de codificación vectores de movimiento.

Obsérvese que en la presente forma de realización, se ha explicado un caso en el que un macrobloque se codifica cada 16 píxeles horizontales x 16 verticales, la compensación de movimiento se realiza en cada bloque de 8 píxeles horizontales x 8 verticales, y una forma de imagen de error de predicción de bloque se codifica cada 8 píxeles
15 horizontales x 8 verticales, pero este procesamiento puede realizarse en otras unidades de píxeles.

Además, en la presente forma de realización, se ha explicado un caso en el que se usa una mediana calculada a partir de vectores de movimiento de tres bloques vecinos codificados previamente como un valor predictivo para codificar un vector de movimiento, si bien puede aplicarse cualquier otro número de bloques vecinos distinto de tres,
20 y el valor predictivo puede determinarse por cualquier otro procedimiento. Por ejemplo, puede usarse un vector de movimiento de uno situado inmediatamente a la izquierda como un valor predictivo, o puede usarse una media aritmética, en lugar de una mediana.

Además, en la presente forma de realización, las posiciones de los bloques vecinos para codificar un vector de
25 movimiento se han explicado usando la fig. 3 y la fig. 4, aunque pueden aplicarse otras posiciones.

Además, en la presente forma de realización, se ha explicado un procedimiento para codificar un bloque actual que usa vectores de movimiento de otros bloques tomando como ejemplos el modo de salto y los modos directos espacial y temporal, aunque puede usarse cualquier otro procedimiento.

30 Además, en la presente forma de realización, se ha explicado un caso en el que puede usarse una diferencia entre un vector de movimiento de un bloque actual y un valor predictivo obtenido de vectores de movimiento de bloques vecinos de manera que se codifique el vector de movimiento, aunque pueden usarse otros procedimiento distintos que la obtención de una diferencia para codificar el vector de movimiento.

35 Además, en la presente forma de realización, se ha explicado un caso en el que cuando un bloque vecino se codifica en modo directo espacial, se calcula una mediana de vectores de movimiento de tres bloques codificados previamente en la vecindad del bloque vecino y se trata como un vector de movimiento del bloque vecino, si bien puede usarse cualquier otro número de bloques distinto de tres, y puede usarse cualquier otro procedimiento para
40 determinar el vector de movimiento. Por ejemplo, puede usarse un vector de movimiento de un bloque situado inmediatamente a la izquierda como un vector de movimiento de un bloque vecino, o puede usarse una media aritmética, en lugar de una mediana.

Además, en la presente forma de realización, cuando un bloque en una imagen B se codifica en modo directo
45 espacial, se calculan dos vectores de movimiento del bloque en las direcciones directa e inversa, aunque pueden calcularse dos vectores de movimiento sólo en la dirección directa o dos vectores de movimiento sólo en la dirección inversa. En este caso, la imagen B se refiere a dos imágenes sólo en la dirección directa o dos imágenes en la dirección inversa.

50 Además, en la presente forma de realización, se ha explicado un caso en el que una imagen predeterminada se refiere a codificación de una imagen P (una imagen P10 se refiere a codificación de una imagen P13, por ejemplo) y dos imágenes predeterminadas se refieren a codificación de una imagen B (las imágenes P10 y P13 se refieren a codificación de una imagen B11), pero estas imágenes P y B pueden codificarse seleccionando imágenes de referencia para cada macrobloque o bloque entre una pluralidad de imágenes. En tal caso, puede generarse un valor
55 predictivo de un vector de movimiento de la forma mostrada en la fig. 13.

La fig. 13 es un organigrama que muestra que el funcionamiento de la unidad de codificación de vectores de movimiento 117 lleva a la obtención de un valor predictivo de un vector de movimiento de un bloque actual para codificar el vector de movimiento, cuando se seleccionan imágenes de referencia para cada bloque.

En primer lugar, la unidad de codificación de vectores de movimiento 117 especifica tres bloques vecinos codificados previamente de un bloque actual (Etapa S300).

- 5 A continuación, la unidad de codificación de vectores de movimiento 117 valora si cada uno de los bloques vecinos especificados es un bloque vecino Ba que ha sido codificado usando vectores de movimiento de otros bloques o un bloque vecino Bb que ha sido codificado sin usar vectores de movimiento de otros bloques (Etapa S302).

10 Aquí, como para el bloque vecino Ba, la unidad de codificación de vectores de movimiento 117 obtiene información que indica los vectores de movimiento usados para codificar el bloque Ba e imágenes de referencia para el bloque vecino Ba, y trata estos vectores de movimiento usados para codificar el bloque Ba como vectores de movimiento del mismo. Al igual que para el bloque vecino Bb, la unidad de codificación de vectores de movimiento 117 obtiene información que indica vectores de movimiento del bloque vecino Bb e imágenes de referencia para el bloque vecino Bb (Etapa S304).

15 A continuación, la unidad de codificación de vectores de movimiento 117 especifica, entre los tres bloques vecinos, un bloque vecino que se refiere a la imagen a la que se refiere un bloque actual basándose en la información obtenida en la Etapa S304 (Etapa S306), y determina el número de los bloques vecinos especificados (Etapa S308).

20 A continuación, si el número de los bloques vecinos valorados en la Etapa S308 es 1, la unidad de codificación de vectores de movimiento 117 considera que el vector de movimiento del bloque vecino que se refiere a la misma imagen es un valor predictivo del vector de movimiento MV del bloque actual (Etapa S310).

25 Si el número de los bloques vecinos valorados en la Etapa S308 no es 1, la unidad de codificación de vectores de movimiento 117 considera que los vectores de movimiento de los bloques vecinos que se refieren a una imagen distinta a la que se refiere el bloque actual, entre los tres bloques vecinos, son 0 (Etapa S312), y considera que una mediana de los vectores de movimiento de los tres bloques vecinos es un valor predictivo del vector de movimiento MV del bloque actual (Etapa S314).

30 Usando el valor predictivo obtenido en la Etapa S310 o la Etapa S314 según se menciona anteriormente, la unidad de codificación de vectores de movimiento 117 calcula una diferencia entre el valor predictivo y el vector de movimiento MV del bloque actual, y codifica la diferencia (Etapa S316).

35 Además, cuando un vector de movimiento se codifica usando un vector de movimiento de un bloque adyacente espacialmente como un valor predictivo, debe almacenarse una cantidad de vectores de movimiento de 1 línea de macrobloque (una parte de 1 macrobloque alta y una anchura de pantalla) en la unidad de almacenamiento de vectores de movimiento 116 para codificar el vector de movimiento, si los vectores de movimiento que han sido usados realmente para compensación de movimiento en modo de salto o modo directo se almacenan en la unidad de almacenamiento de vectores de movimiento 116. Esto se aplica al caso en que los vectores de movimiento que han sido usados realmente para compensación de movimiento en modo de salto o modo directo se almacenan en la unidad de almacenamiento de vectores de movimiento 116. Esta es la razón por la que cuando se usan los bloques vecinos explicados en relación con la fig. 3 y la fig. 4 de la presente forma de realización, existen pasado 1 macrobloque cortes de bloques que se refieren como bloques vecinos para codificar el vector de movimiento, con el macrobloque actual como punto de partida.

45 **(Segunda forma de realización)**

Se explicará un aparato de decodificación de imágenes en movimiento 700 en la segunda forma de realización de la presente invención con referencia a las figuras.

50 La fig. 14 es un diagrama de bloques que muestra la estructura del aparato de decodificación de imágenes en movimiento 700 en la segunda forma de realización de la presente invención.

55 El aparato de decodificación de imágenes en movimiento 700 según se muestra en la fig. 14 decodifica imágenes en movimiento codificadas por el aparato de codificación de imágenes en movimiento 100 en la primera forma de realización, e incluye una unidad de análisis de flujo de bits 701, una unidad de decodificación de errores de predicción 702, una unidad de decodificación de modo 703, una unidad de decodificación de compensación de movimiento 705, una unidad de almacenamiento de vectores de movimiento 706, una memoria de trama 707, una unidad de suma 708, conmutadores 709 y 710 y una unidad de decodificación de vectores de movimiento 711.

La unidad de análisis de flujo de bits 701 extrae varios datos del flujo de bits de entrada. Aquí, varios datos incluyen información sobre el modo de codificación, información sobre vectores de movimiento, y así sucesivamente. La información extraída de modo de codificación se envía a la unidad de decodificación de modo 703. La información
 5 extraída de vectores de movimiento se envía a la unidad de decodificación de vectores de movimiento 711. Además, los datos extraídos de errores de predicción codificados se envían a la unidad de decodificación de errores de predicción 702.

La unidad de decodificación de errores de predicción 702 decodifica los datos de errores de predicción codificados
 10 introducidos para generar una forma de imagen de error de predicción. La forma de imagen de error de predicción generada se envía al conmutador 709. Cuando el conmutador 709 se conecta al terminal "b", la forma de imagen de error de predicción se envía a la unidad de suma 708.

La unidad de decodificación de modo 703 controla el conmutador 709 y el conmutador 710 con referencia a la
 15 información de modo de codificación extraída del flujo de bits. Si el modo de codificación es codificación intraimagen, la unidad de decodificación de modo 703 controla los conmutadores 709 y 710 para conectarse al terminal "a" y el terminal "c", respectivamente, y si el modo de codificación es codificación entre imágenes, controla los conmutadores 709 y 710 para conectarse al terminal "b" y el terminal "d", respectivamente. La unidad de
 20 decodificación de modo 703 envía además la información del modo de codificación a la unidad de decodificación de vectores de movimiento 711.

La unidad de decodificación de vectores de movimiento 711 decodifica la información de vectores de movimiento enviado desde la unidad de análisis de flujo de bits 701.

25 Para ser más concreto, cuando la información del modo de codificación indica codificación de predicción entre imágenes usando vectores de movimiento, la unidad de decodificación de vectores de movimiento 711 obtiene un valor predictivo para un bloque actual que se decodificará usando los vectores de movimiento de bloques vecinos previamente decodificados, de la misma manera que se describe en la fig. 3 y la fig. 4. Por ejemplo, según se muestra en la fig. 3, la unidad de decodificación de vectores de movimiento 711 obtiene un valor predictivo para un
 30 bloque actual A del vector de movimiento MVb del bloque vecino B, el vector de movimiento MVC del bloque vecino C y el vector de movimiento MVd del bloque vecino D. Aquí, el valor predictivo se calcula basándose en una mediana calculada a partir de cada una de las componentes horizontales y las componentes verticales de los tres vectores de movimiento previamente decodificados MVb, MVC y MVd. A continuación, la unidad de decodificación de vectores de movimiento 711 añade el valor predictivo a la diferencia que es la información del vector de movimiento
 35 enviada desde la unidad de análisis de flujo de bits 701 de manera que se determine el vector de movimiento MV del bloque actual A. Cuando la información del modo de codificación es cualquiera de los anteriormente mencionados modo de salto, modo directo temporal y modo directo espacial, la unidad de decodificación de vectores de movimiento 711 determina el vector de movimiento usando sólo los vectores de movimiento de los bloques vecinos previamente decodificados.

40 La fig. 15 es un organigrama que muestra el funcionamiento general de la unidad de decodificación de vectores de movimiento 711 en la presente forma de realización.

En primer lugar, la unidad de decodificación de vectores de movimiento 711 especifica tres bloques vecinos
 45 previamente decodificados de un bloque actual que se decodificará (Etapa S200).

A continuación, la unidad de decodificación de vectores de movimiento 711 valora si cada uno de los bloques vecinos especificados es un bloque vecino que ha sido codificado usando vectores de movimiento de otros bloques o un bloque vecino Bb que ha sido codificado sin usar vectores de movimiento de otros bloques (Etapa S202).
 50

Como resultado, la unidad de decodificación de vectores de movimiento 711 determina si un bloque vecino Ba está incluido o no en los tres bloques vecinos especificados (Etapa S204).

55 Cuando se valora en la Etapa S204 que un bloque vecino Ba está incluido (Y en la Etapa S204), la unidad de decodificación de vectores de movimiento 711 obtiene un valor predictivo de los vectores de movimiento de los tres bloques vecinos tratando un vector de movimiento calculado a partir de vectores de movimiento de otros bloques para decodificar el bloque vecino Ba como un vector de movimiento del bloque vecino Ba, según se menciona anteriormente (Etapa S206).

Por otra parte, cuando se valora en la Etapa S206 que un bloque vecino Ba no está incluido (N en la Etapa S204), la unidad de decodificación de vectores de movimiento 711 obtiene un valor predictivo de los vectores de movimiento obtenidos respectivamente basándose en los resultados de la estimación de los tres bloques vecinos Bb (Etapa S208).

5

A continuación, la unidad de decodificación de vectores de movimiento 711 suma el valor predictivo obtenido en la Etapa S206 o S208 a la diferencia que es la información de vectores de movimiento enviada desde la unidad de análisis de flujo de bits 701, de manera que se decodifique el vector de movimiento codificado del bloque actual (Etapa S210). La unidad de decodificación de vectores de movimiento 711 también envía el vector de movimiento

10 decodificado a la unidad de decodificación de compensación de movimiento 705.

La unidad de almacenamiento de vectores de movimiento 706 almacena el vector de movimiento decodificado en la unidad de decodificación de vectores de movimiento 711 y el modo de codificación obtenido en la unidad de decodificación de modo 703.

15

La unidad de decodificación de compensación de movimiento 705 obtiene una forma de imagen de compensación de movimiento de todos los macrobloques de la memoria de trama 707 basándose en el vector de movimiento decodificado en la unidad de decodificación de vectores de movimiento 711.

20 La unidad de suma 708 suma la forma de imagen de error de predicción introducida y la forma de imagen de compensación de movimiento para generar la forma de imagen decodificada, y produce la forma de imagen decodificada generada para la memoria de trama 707.

La memoria de trama 707 almacena la forma de imagen decodificada generada por la unidad de suma 708 según
25 todas las bases de imágenes.

En primer lugar se explicará el funcionamiento de este aparato de decodificación de imágenes en movimiento 700, en particular el funcionamiento general del mismo.

30 La fig. 16 es una ilustración para explicar la entrada y salida del aparato de decodificación de imágenes en movimiento 700.

Según se muestra en la fig. 16A, el aparato de decodificación de imágenes en movimiento 700 obtiene el flujo de bits producido desde el aparato de codificación de imágenes en movimiento 100 en la primera forma de realización en orden de salida, y decodifica las imágenes incluidas en el flujo de bits en secuencia. A continuación, según se
35 muestra en la fig. 16B, el aparato de decodificación de imágenes en movimiento 700 reordena las imágenes decodificadas en orden de visualización para salida.

A continuación se explicará el procesamiento de decodificación realizado por el aparato de decodificación de
40 imágenes en movimiento 700 anterior tomando la decodificación de la imagen P13 y la imagen B11 según se muestra en la fig. 16 como un ejemplo específico.

(Decodificación de la imagen P13)

45 En primer lugar, la unidad de análisis de flujo de bits 701 del aparato de decodificación de imágenes en movimiento 700 obtiene el flujo de bits con respecto a la imagen P13, y extrae la información de selección de modo y la información de vectores de movimiento y los datos de errores de predicción codificados del flujo de bits.

La unidad de decodificación de modo 703 controla los conmutadores 709 y 710 con referencia a la información de
50 selección de modo extraída del flujo de bits de la imagen P13.

A continuación se explicará un caso en el que la información de selección de modo indica codificación de predicción entre imágenes.

55 La unidad de decodificación de vectores de movimiento 711 realiza el procesamiento de decodificación anterior en la información de vectores de movimiento extraída del flujo de bits de la imagen P13 sobre una base de bloque por bloque basándose en la información de selección de modo que indica la codificación de predicción entre imágenes producidos de la unidad de decodificación de modo 703.

Aquí, cuando se decodifica el vector de movimiento del bloque actual en la imagen P13, la unidad de decodificación de vectores de movimiento 711 especifica tres bloques vecinos decodificados previamente del bloque actual, y valora si cada uno de estos bloques vecinos ha sido codificado usando o no vectores de movimiento de otros bloques. Cuando cualquiera de los bloques vecinos es un bloque que ha sido codificado usando vectores de movimiento de otros bloques, es decir, en modo de salto, la unidad de decodificación de vectores de movimiento 711 trata un vector de movimiento calculado a partir de los vectores de movimiento de los otros bloques para decodificar el bloque vecino como un vector de movimiento del bloque vecino, de la misma manera que hace la unidad de codificación de vectores de movimiento 117 en la primera forma de realización. Para ser más concreto, la unidad de decodificación de vectores de movimiento 711 calcula la mediana de los vectores de movimiento de los tres bloques codificados previamente en la vecindad de ese bloque vecino, y lo trata como un vector de movimiento del bloque vecino.

Además, la unidad de almacenamiento de vectores de movimiento 706 almacena la información de selección de modo producida desde la unidad de decodificación de modo 703, y la unidad de decodificación de vectores de movimiento 711 valora si cada uno de los bloques vecinos es o no un bloque que ha sido codificado usando vectores de movimiento de otros bloques basándose en la información de selección de modo almacenado en la unidad de almacenamiento de vectores de movimiento 706. La unidad de almacenamiento de vectores de movimiento 706 almacena además los vectores de movimiento de los otros bloques usados para decodificar el bloque vecino. Para ser más concreto, la unidad de almacenamiento de vectores de movimiento 706 almacena los vectores de movimiento de los tres bloques en la vecindad del bloque vecino que ha sido codificado en modo de salto. Cuando se decodifica el vector de movimiento del bloque actual, la unidad de decodificación de vectores de movimiento 711 calcula una mediana a partir de los vectores de movimiento de los tres bloques anteriores almacenados en la unidad de almacenamiento de vectores de movimiento 706. Obsérvese que la unidad de almacenamiento de vectores de movimiento 706 puede almacenar con antelación un vector de movimiento de un bloque que ha sido codificado usando vectores de movimiento de otros bloques, calculando una mediana de los vectores de movimiento para decodificar el bloque. En este caso, cuando se decodifica el vector de movimiento del bloque actual, la unidad de decodificación de vectores de movimiento 711 no necesita obtener el vector de movimiento del bloque vecino que ha sido codificado en modo de salto, sino que puede usar el vector de movimiento almacenado en la unidad de almacenamiento de vectores de movimiento 706 directamente como un vector de movimiento del bloque vecino.

Por otra parte, los datos de errores de predicción codificados del macrobloque actual de la imagen P13 se decodifica en la unidad de decodificación de errores de predicción 702 y se genera como una forma de imagen de error de predicción, y los conmutadores 709 y 710 se conectan a la unidad de suma 708. Por tanto, la forma de imagen de compensación de movimiento generada basándose en el vector de movimiento decodificado en la unidad de decodificación de vectores de movimiento 711 se suma a la forma de imagen de error de predicción y se envía a la memoria de trama 707.

Además, cuando se decodifica un vector de movimiento de una imagen P, la unidad de decodificación de vectores de movimiento 711 almacena su vector de movimiento y un modo de codificación obtenido de la unidad de decodificación de modo 703 en la unidad de almacenamiento de vectores de movimiento 706 para decodificar las siguientes imágenes y bloques.

Los restantes macrobloques de la imagen P13 se decodifican en secuencia. Después de completar la decodificación de todos los macrobloques de la imagen P13, sigue la decodificación de la imagen B11.

(Decodificación de la imagen B11)

En primer lugar, la unidad de análisis de flujo de bits 701 del aparato de decodificación de imágenes en movimiento 700 obtiene el flujo de bits de la imagen B11, y extrae la información de selección de modo y la información de vectores de movimiento y los datos de errores de predicción codificados a partir del flujo de bits.

La unidad de decodificación de modo 703 controla los conmutadores 709 y 710 con referencia a la información de selección de modo extraída del flujo de bits de la imagen B11.

A continuación se explicará un caso en el que la información de selección de modo indica la codificación de predicción entre imágenes.

La unidad de decodificación de vectores de movimiento 711 realiza el procesamiento de decodificación anterior sobre la información de vectores de movimiento extraída del flujo de bits de la imagen B11 en una base de bloque

por bloque basándose en la información de selección de modo que indica la codificación de predicción entre las imágenes producidas desde la unidad de decodificación de modo 703.

5 Cuando se decodifica un vector de movimiento de un bloque actual en la imagen B11, la unidad de decodificación de vectores de movimiento 711 especifica tres bloques vecinos decodificados previamente del bloque actual, y valora si cada uno de estos bloques vecinos ha sido codificado usando o no vectores de movimiento de otros bloques. Cuando cualquiera de los bloques vecinos es un bloque que ha sido codificado usando vectores de movimiento de otros bloques, es decir, en modo directo temporal o espacial, la unidad de decodificación de vectores de movimiento 711 trata un vector de movimiento obtenido usando los vectores de movimiento de los otros bloques para decodificar el bloque vecino como un vector de movimiento del mismo, de la misma manera que hace la unidad de codificación de vectores de movimiento 117 en la primera forma de realización.

Más específicamente, cuando el bloque vecino ha sido codificado en modo directo temporal, la unidad de decodificación de vectores de movimiento 711 lee desde la unidad de almacenamiento de vectores de movimiento 706 un vector de movimiento de un bloque, que se sitúa al lado de un bloque vecino que ha sido codificado en modo directo, en una imagen de referencia recién decodificada previamente (imagen P13). Por ejemplo, según se muestra en la fig. 11, si el bloque vecino C ha sido codificado en modo directo temporal, la unidad de decodificación de vectores de movimiento 711 lee el vector de movimiento decodificado del bloque X en la imagen P13 a partir de la unidad de almacenamiento de vectores de movimiento 706. A continuación, la unidad de decodificación de vectores de movimiento 711 calcula a un vector de movimiento en sentido directo MVFc y un vector de movimiento en sentido inverso MVBc usados para codificar el bloque vecino C usando la Ecuación 1 y la Ecuación 2, y usa estos vectores de movimiento MVFc y MVBc como vectores de movimiento del bloque vecino C.

En el caso anterior, la unidad de decodificación de vectores de movimiento 711 lee a partir de la unidad de almacenamiento de vectores de movimiento 706 el vector de movimiento MVP del bloque X en la imagen P13 que se sitúa al lado del bloque vecino C que ha sido codificado en modo directo. Sin embargo, en cuanto a un bloque que ha sido codificado usando vectores de movimiento de otros bloques, la unidad de almacenamiento de vectores de movimiento 706 puede almacenar el vector de movimiento del bloque calculado a partir de los vectores de movimiento de los otros bloques para decodificar el bloque. En este caso, la unidad de almacenamiento de vectores de movimiento 706 almacena los vectores de movimiento MVFc y MVBc con antelación. Por tanto, cuando se decodifica el vector de movimiento del bloque actual A, la unidad de decodificación de vectores de movimiento 711 no necesita calcular los vectores de movimiento MVFc y MVBc para el bloque vecino C leyendo el vector de movimiento MVP del bloque X y usando la Ecuación 1 y la Ecuación 2, sino que puede usar los vectores de movimiento MVFc y MVBc almacenados en la unidad de almacenamiento de vectores de movimiento 706 directamente como vectores de movimiento del bloque vecino C.

Por otra parte, cuando un bloque vecino ha sido codificado en modo directo espacial, la unidad de decodificación de vectores de movimiento 711 trata los vectores de movimiento calculados usando vectores de movimiento de otros bloques en la vecindad del bloque vecino como vectores de movimiento de los mismos. Por ejemplo, en la situación que se muestra en la fig. 12, la unidad de decodificación de vectores de movimiento 711 calcula medianas a partir de los vectores de movimiento de los tres bloques decodificados previamente E, F y G en la vecindad del bloque vecino C que ha sido codificado en modo directo espacial, y trata el vector de movimiento en sentido inverso MVFc y el vector de movimiento en sentido directo MVBc indicados por las medianas como vectores de movimiento del bloque vecino C.

Además, la unidad de almacenamiento de vectores de movimiento 706 almacena vectores de movimiento usados para decodificar un bloque que ha sido codificado sin usar vectores de movimiento de otros bloques. Para ser más concreto, en la situación que se muestra en la fig. 12, la unidad de almacenamiento de vectores de movimiento 706 almacena los vectores de movimiento de los tres bloques E, F y G en la vecindad del bloque vecino C que ha sido codificado en modo directo espacial. Por tanto, cuando se decodifica el vector de movimiento del bloque actual A, la unidad de decodificación de vectores de movimiento 711 calcula los vectores de movimiento MVFc y MVBc para el bloque vecino a partir de los vectores de movimiento de los tres bloques E, F y G anteriores almacenados en la unidad de almacenamiento de vectores de movimiento 706. Obsérvese que la unidad de almacenamiento de vectores de movimiento 706 puede almacenar con antelación vectores de movimiento obtenidos calculando medianas para decodificar un bloque que ha sido codificado usando vectores de movimiento de otros bloques. En este caso, en la situación que se muestra en la fig. 12, la unidad de almacenamiento de vectores de movimiento 706 almacena los vectores de movimiento MVFc y MVBc con antelación. Por tanto, cuando se decodifican los vectores de movimiento del bloque actual A, la unidad de decodificación de vectores de movimiento 711 no necesita calcular los vectores de movimiento del bloque vecino C que ha sido codificado en modo directo espacial, sino que puede

usar los vectores de movimiento MVFc y MVBc almacenados en la unidad de almacenamiento de vectores de movimiento 706 directamente como vectores de movimiento del bloque vecino C.

5 Aquí, cuando se decodifican los vectores de movimiento de un bloque actual que ha de decodificarse, si el bloque vecino decodificado previamente del bloque actual ha sido procesado en codificación intraimagen, no en codificación entre imágenes según se menciona anteriormente, el aparato de decodificación de imágenes en movimiento 700 realiza un procesamiento excepcional.

10 Por ejemplo, cuando uno de los tres bloques vecinos se ha sometido a codificación intraimagen, la unidad de decodificación de vectores de movimiento 711 del aparato de decodificación de imágenes en movimiento 700 realiza el procesamiento considerando que los vectores de movimiento del bloque vecino son 0. Cuando dos bloques vecinos se han sometido a codificación intraimagen, la unidad de decodificación de vectores de movimiento 711 usa los vectores de movimiento del bloque vecino restante como valores predictivos de los vectores de movimiento del bloque actual. Además, cuando los tres bloques vecinos se han sometido a codificación intraimagen, la unidad de
15 decodificación de vectores de movimiento 711 decodifica los vectores de movimiento del bloque actual considerando que los valores predictivos de los mismos son 0.

Por otra parte, los datos de errores de predicción codificados para el macrobloque actual de la imagen B11 han sido decodificados en la unidad de decodificación de errores de predicción 702 y generados como una forma de imagen de error de predicción, y los conmutadores 709 y 710 están conectados a la unidad de suma 708. Por tanto, la forma
20 de imagen de compensación de movimiento generada basándose en el vector de movimiento decodificado por la unidad de decodificación de vectores de movimiento 711 se suma a la forma de imagen de error de predicción y se envía a la memoria de trama 707.

25 El procesamiento de decodificación de los restantes macrobloques de la imagen B11 se realiza de la misma manera. Después de completar el procesamiento para todos los macrobloques de la imagen B11, sigue el procesamiento de decodificación de la imagen B12.

Según se describe anteriormente, según el procedimiento de decodificación de vectores de movimiento de la presente invención, se obtiene un valor predictivo a partir de los vectores de movimiento de bloques codificados vecinos decodificados previamente, y se decodifica un vector de movimiento de cada bloque actual usando el valor
30 predictivo y la diferencia. Si alguno de los bloques vecinos ha sido codificado usando vectores de movimiento de otros bloques, por ejemplo, en modo de salto o modo directo, un valor predictivo se obtiene usando, como vector de movimiento del bloque vecino, un vector de movimiento calculado a partir de los vectores de movimiento de los otros
35 bloques para decodificación de ese bloque vecino.

En consecuencia, los vectores de movimiento que han sido codificados de la manera que se muestra en la primera forma de realización pueden decodificarse apropiadamente.

40 Obsérvese que, en la presente forma de realización, se ha explicado un caso en el que se usa una mediana calculada a partir de vectores de movimiento de tres bloques vecinos decodificados previamente como un valor predictivo para decodificar un vector de movimiento, si bien puede aplicarse cualquier otro número de bloques vecinos distinto a tres, y el valor predictivo puede determinarse mediante cualquier otro procedimiento. Por ejemplo, puede usarse un vector de movimiento de un bloque inmediatamente a la izquierda como valor predictivo, o una
45 media aritmética, en lugar de una mediana.

Además, en la presente forma de realización, las posiciones de bloques vecinos para decodificar un vector de movimiento se han explicado usando la fig. 3 y la fig. 4, aunque puede aplicarse cualquier otra posición.

50 Además, en la presente forma de realización, se ha explicado un procedimiento para codificar un bloque actual usando vectores de movimiento de otros bloques tomando como ejemplos modo de salto y modos directos temporal y espacial, aunque puede usarse cualquier otro modo.

Además, en la presente forma de realización, se ha explicado un caso en el que un vector de movimiento es
55 decodificado sumando un valor predictivo obtenido de vectores de movimiento de bloques vecinos y una diferencia según se indica en un flujo de bits, si bien puede usarse cualquier otro procedimiento que la suma para decodificar el vector de movimiento.

Además, en la presente forma de realización, se ha explicado un caso en el que cuando un bloque vecino ha sido

codificado en modo directo espacial, se calcula una mediana de vectores de movimiento de tres bloques codificados previamente en la vecindad del bloque vecino y se trata como un vector de movimiento del bloque vecino, aunque puede usarse cualquier otro número de bloques distinto a tres, y puede usarse cualquier otro procedimiento para determinar el vector de movimiento. Por ejemplo, puede usarse un vector de movimiento de un bloque inmediatamente a la izquierda como vector de movimiento del bloque vecino, o una media aritmética, en lugar de una mediana.

Además, en la presente forma de realización, cuando existe un bloque vecino que ha sido codificado en modo directo espacial, se calculan dos vectores de movimiento del bloque en las direcciones directa e inversa, si bien pueden calcularse dos vectores de movimiento sólo en la dirección directa o dos vectores de movimiento sólo en la dirección inversa. En este caso, una imagen B actual para decodificación se refiere a dos imágenes sólo en la dirección directa o dos imágenes sólo en la dirección inversa.

Además, en la presente forma de realización, se ha explicado un caso en el que se refiere una imagen predeterminada a decodificación de una imagen P (la imagen P10 se refiere a decodificación de la imagen P13, por ejemplo) y se refieren dos imágenes predeterminadas a decodificación de una imagen B (las imágenes P10 y P13 se refieren a decodificación de la imagen B11), pero esta imagen P y esta imagen B pueden decodificarse seleccionando imágenes de referencia de entre una pluralidad de imágenes en cada base de macrobloques o bloques. En tal caso, puede generarse un valor predictivo de un vector de movimiento de la manera que se muestra en la fig. 17.

La fig. 17 es un organigrama que muestra el funcionamiento de la unidad de decodificación de vectores de movimiento 711 para obtener un valor predictivo de un vector de movimiento de un bloque actual que se decodificará y decodificar el vector de movimiento usando el valor predictivo cuando una imagen de referencia se selecciona según una base de bloque por bloque.

En primer lugar, la unidad de decodificación de vectores de movimiento 711 especifica tres bloques vecinos decodificados previamente del bloque actual (Etapa S400).

A continuación, la unidad de decodificación de vectores de movimiento 711 valora si cada uno de los bloques vecinos especificados es un bloque vecino Ba que ha sido codificado usando vectores de movimiento de otros bloques, o un bloque vecino Bb que ha sido codificado sin usar vectores de movimiento de otros bloques (Etapa S402).

Aquí, como para el bloque vecino Ba, la unidad de decodificación de vectores de movimiento 711 obtiene información que indica un vector de movimiento usado para decodificar el bloque vecino Ba y la imagen de referencia a la que se refiere, y trata el vector de movimiento usado para la decodificación como un vector de movimiento del bloque vecino Ba. Al igual que para el bloque vecino Bb, la unidad de decodificación de vectores de movimiento 711 obtiene información que indica el vector de movimiento del bloque vecino Bb y la imagen de referencia a la que se refiere (Etapa S404).

A continuación, la unidad de decodificación de vectores de movimiento 711 especifica el bloque vecino que se refiere a la imagen a la que se refiere el bloque actual, entre los tres bloques vecinos, basándose en la información obtenida en la Etapa S404 (Etapa S406), y determina el número de los bloques vecinos especificados (Etapa S408).

Si el número de los bloques vecinos determinado en la Etapa S408 es 1, la unidad de decodificación de vectores de movimiento 711 considera el vector de movimiento de ese bloque vecino que se refiere a la misma imagen que será un valor predictivo del vector de movimiento del bloque actual (Etapa S410).

Si el número de los bloques vecinos determinados en la Etapa S408 es un número distinto a uno, la unidad de decodificación de vectores de movimiento 711 considera que el vector de movimiento del bloque vecino, entre los tres bloques vecinos, que se refiere a otra imagen distinta a la que se refiere el bloque actual es 0 (Etapa S412), y considera la mediana de los vectores de movimiento de los tres bloques vecinos como un valor predictivo del vector de movimiento del bloque actual (Etapa S414).

Según se describe anteriormente, el vector de movimiento codificado del bloque actual se decodifica sumando la diferencia al valor predictivo obtenido en la Etapa S410 o la Etapa S414.

Además, cuando un vector de movimiento se decodifica usando un vector de movimiento de un bloque adyacente

espacialmente como un valor predictivo, debe almacenarse una cantidad de vectores de movimiento de 1 línea de macrobloques (una parte de 1 macrobloque de alto y una pantalla de ancho) en la unidad de almacenamiento de vectores de movimiento 706 para decodificar el vector de movimiento, si los vectores de movimiento que se han usado realmente para compensación de movimiento en modo de salto o modo directo se almacenan en la unidad de almacenamiento de vectores de movimiento 706. Esto se aplica al caso en el que los vectores de movimiento que se han usado realmente para compensación de movimiento en modo de salto o modo directo se almacenan en la unidad de almacenamiento de vectores de movimiento 706. Esta es la razón por la que cuando se usan los bloques vecinos explicados en relación con las fig. 3 y la fig. 4 de la presente forma de realización, existe 1 corte de bloques pasado 1 macrobloque que se refiere como bloques vecinos para decodificar el vector de movimiento, con el macrobloque actual como punto de partida.

(Tercera forma de realización)

Además, si se graba un programa para realizar el procedimiento de codificación de vectores de movimiento o el procedimiento de decodificación de vectores de movimiento según se muestra en cada una de las formas de realización mencionadas anteriormente en un medio de almacenamiento como un disco flexible, se hace posible realizar el procesamiento según se muestra en cada una de las formas de realización anteriores fácilmente en un sistema informático independiente.

La fig. 18 es una ilustración de un medio de almacenamiento que almacena un programa para realizar el procedimiento de codificación de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento ejecutado por el aparato de codificación de imágenes en movimiento 100 en la primera forma de realización y el aparato de decodificación de imágenes en movimiento 200 en la segunda forma de realización por un sistema informático.

La fig. 18B muestra la vista frontal y la vista en sección transversal de la apariencia de un disco flexible FD, y un disco FD1, y la fig. 18A muestra un ejemplo de un formato físico del disco FD1 como un medio de grabación en sí.

El disco FD1 está contenido en una caja F, se forma una pluralidad de pistas Tr concéntricamente en la superficie del disco FD1 en la dirección radial desde la periferia, y cada pista se divide en 16 sectores Se en la dirección angular. Por tanto, en el disco flexible que almacena el programa mencionado anteriormente, el procedimiento de codificación de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento como el programa anterior se graban en un área asignada para él en el disco FD1.

La fig. 18C muestra la estructura para grabar y reproducir el programa en y desde el disco flexible FD.

Para grabar el programa en el disco flexible FD, el sistema informático Cs escribe el procedimiento de codificación de vectores de movimiento o el procedimiento de decodificación de vectores de movimiento como el programa en el disco flexible FD por medio de un lector de discos flexibles FDD. Para construir el procedimiento de codificación anterior de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento en el sistema informático Cs mediante el programa grabado en el disco flexible FD, el programa es leído desde el disco flexible FD por medio del lector de discos flexibles FDD y transferido al sistema informático Cs.

Obsérvese que la explicación anterior se realiza según el supuesto de que un medio de grabación es un disco flexible FD, pero el mismo procesamiento puede realizarse también usando un disco óptico. Además, el medio de grabación no se limita a éstos, sino que puede usarse de la misma manera cualquier otro medio como una tarjeta CI o una casete ROM si es posible grabar un programa en ellos.

(Cuarta forma de realización)

Además, se explicarán aquí las aplicaciones del procedimiento de codificación de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento según se muestra en las formas de realización anteriores y un sistema que los usa.

La fig. 19 es un diagrama de bloques que muestra la configuración general de un sistema de suministro de contenidos ex100 para realizar el servicio de distribución de contenidos. El área para proporcionar el servicio de comunicación está dividida en celdas del tamaño deseado, y se colocan estaciones de base ex107~ex110 que son estaciones inalámbricas fijas en celdas respectivas.

En este sistema de suministro de contenidos ex100, se conectan varios dispositivos como un ordenador ex111, una PDA (asistente digital personal) ex112, una cámara ex113, un teléfono móvil ex114 y un teléfono móvil equipado con cámara ex115 a Internet ex101, por medio de un proveedor de servicios de Internet ex102, una red telefónica ex104 y estaciones de base ex107~ex110, por ejemplo.

5

Sin embargo, el sistema de suministro de contenidos ex100 no se limita a la combinación según se muestra en la fig. 19, y puede conectarse a una combinación de cualquiera de ellos. Además, cada dispositivo puede conectarse directamente a la red telefónica ex104, no a través de las estaciones de base ex107~ex110 que son las estaciones inalámbricas fijas.

10

La cámara ex113 es un dispositivo como una videocámara digital capaz de obtener imágenes en movimiento. El teléfono móvil puede ser cualquiera entre un teléfono móvil de un sistema PDC (Personal Digital Communications), un sistema CDMA (Code Multiple Division Access), un sistema W-CDMA (Wideband-Code Multiple Division Access) o un sistema GSM (Global System for Mobile Communications), un PHS (Personal Handyphone System) y similares.

15

Además, se conecta un servidor de flujo continuo ex103 a la cámara ex113 por medio de la estación de base ex109 y la red telefónica ex104, lo que permite la distribución en directo o similar usando la cámara ex113 basándose en los datos codificados emitidos desde el usuario. La cámara ex113 o el servidor para emitir los datos pueden codificar las tomas de datos de la cámara. Además, la toma de datos de imágenes en movimiento por una cámara ex116

20

puede emitirse al servidor de flujo continuo ex103 por medio del ordenador ex111. La cámara ex116 es un dispositivo como una cámara digital capaz de obtener imágenes fijas y en movimiento. En este caso, la cámara ex116 o el ordenador ex111 pueden codificar los datos de las imágenes en movimiento. Un LSI ex117 incluido en el ordenador ex111 o la cámara ex116 realiza el procesamiento de codificación. Obsérvese que el software para codificación y decodificación de las imágenes puede estar integrado en cualquier tipo de un medio de

25

almacenamiento (como un CD-ROM, un disco flexible y un disco duro) que es un medio de grabación que puede ser leído por el ordenador ex111 o similar. Además, el teléfono móvil provisto de cámara ex115 puede emitir los datos de imágenes en movimiento. Estos datos de imágenes en movimiento son los datos codificados por el LSI incluido en el teléfono móvil ex115.

30

En esta sistema de suministro de contenidos ex100, los contenidos (como un vídeo de música en directo) tomados por los usuarios usando la cámara ex113, la cámara ex116 o similar se codifican de la misma manera que las formas de realización anteriores y se emiten al servidor de flujo continuo ex103, mientras el servidor de flujo continuo ex103 realiza distribución continua de los datos del contenido anterior a los clientes a petición de éstos. Los clientes incluyen el ordenador ex111, la PDA ex112, la cámara ex113, el teléfono móvil ex114 y demás elementos capaces

35

de decodificar los datos codificados mencionados anteriormente. El sistema de suministro de contenidos ex100 es un sistema en el que los clientes pueden así recibir y reproducir los datos codificados, y además pueden recibir, decodificar y reproducir los datos en tiempo real de manera que pueda realizarse una difusión personal.

40

Cuando cada dispositivo de este sistema realiza codificación o decodificación, puede usarse el aparato de codificación de imágenes en movimiento o el aparato de decodificación de imágenes en movimiento según se muestra en cada una de las formas de realización mencionadas anteriormente.

Se explicará un teléfono móvil como un ejemplo de las mismas.

45

La fig. 20 es un diagrama que muestra un teléfono móvil ex115 que usa el procedimiento de codificación de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento según se explica en las formas de realización anteriores. El teléfono móvil ex115 tiene una antena ex201 para enviar y recibir ondas de radio entre la estación de base ex110, una unidad de cámara ex203 como una cámara CCD capaz de tomar imágenes de vídeo y fijas, una unidad de pantalla ex202 como una pantalla de cristal líquido para mostrar los datos obtenidos por tomas

50

de vídeo de decodificación por parte de la unidad de cámara ex203, vídeo recibido por la antena ex201, o similar, un cuerpo principal que incluye un conjunto de teclas de funcionamiento ex204, una unidad de salida de voz ex208 como un altavoz para producir voces, una unidad de entrada de voz ex205 como un micrófono para introducir voces, un medio de almacenamiento ex207 para almacenar datos codificados o decodificados, como datos de imágenes en movimiento o fijas tomadas por la cámara, y datos de texto, imágenes en movimiento o imágenes fijas de mensajes

55

de correo electrónico recibidos, y una unidad de ranura ex206 para fijar el medio de almacenamiento ex207 en el teléfono móvil ex115. El medio de almacenamiento ex207 incluye un elemento de memoria flash, una clase de EEPROM (Memoria de Sólo Lectura Programable y Borrable Eléctricamente) que es una memoria no volátil borrrable y reescribible eléctricamente, en una caja de plástico como una tarjeta SD.

Además, el teléfono móvil ex115 se explicará con referencia a la fig. 21. En el teléfono móvil ex115, se conecta una unidad de control principal ex311 para control global de cada unidad del cuerpo principal que incluye la unidad de pantalla ex202 y las teclas de funcionamiento ex204 a una unidad de circuito de fuente de alimentación ex310, una unidad de control de entrada de operación ex304, una unidad de codificación de la imagen ex312, una unidad de interfaz de cámara ex303, una unidad de control LCD (Pantalla de Cristal Líquido) ex302, una unidad de comunicación de imágenes ex309, una unidad de multiplexado/demultiplexado ex308, una unidad de grabación/reproducción ex307, una unidad de circuito de módem ex306 y una unidad de tratamiento de voz ex305 entre sí por medio de un bus síncrono ex313.

- 10 Cuando una tecla de fin de llamada o de potencia se ACTIVA por la acción de un usuario, la unidad de circuito de fuente de alimentación ex310 suministra a las unidades respectivas potencia desde una batería de manera que se activa el teléfono móvil digital provisto de cámara ex115 para un estado estacionario.

- En el teléfono móvil ex115, bajo el control de la unidad de control principal ex311 que incluye una CPU, ROM, RAM y similar, la unidad de tratamiento de voz ex305 convierte las señales de voz recibidas por la unidad de entrada de voz ex205 en modo de conversación en datos de voz digital, la unidad de circuito de módem ex306 realiza procesamiento de espectro ensanchado de los datos de voz digital, y la unidad de circuitos de envío/recepción ex301 realiza conversión digital-analógica y transformación de frecuencia de los datos, de manera que se transmita el resultado por medio de la antena ex201. Además, en el teléfono móvil ex115, los datos recibidos por la antena ex201 en modo conversación se amplifican y se someten a transformación de frecuencia y conversión analógico-digital, la unidad de circuito de módem ex306 realiza procesamiento inverso de espectro ensanchado de los datos, y la unidad de tratamiento de voz ex305 la convierte en datos de voz analógicos, de manera que se emita el resultado por medio de la unidad de salida de voz ex208.

- Además, cuando se envía un mensaje de correo electrónico en modo de comunicación de datos, los datos de texto del mensaje de correo electrónico introducido accionando las teclas de funcionamiento ex204 en el cuerpo principal son enviados a la unidad de control principal ex311 por medio de la unidad de control de entrada de funcionamiento ex304. En la unidad de control principal ex311, después de que la unidad de circuito de módem ex306 realiza procesamiento de espectro ensanchado de los datos de texto y la unidad de circuitos de envío/recepción ex301 realiza conversión digital-analógica y transformación de frecuencia de los mismos, el resultado se envía a la estación de base ex110 por medio de la antena ex201.

- Cuando los datos de imágenes se emiten en modo de comunicación de datos, los datos de imágenes tomados por la unidad de cámara ex203 se proporcionan a la unidad de codificación de la imagen ex312 por medio de la unidad de interfaz de la cámara ex303. Cuando los datos de imágenes no se emiten, los datos de imágenes tomados por la unidad de cámara ex203 pueden visualizarse también directamente en la unidad de pantalla 202 por medio de la unidad de interfaz de la cámara ex303 y la unidad de control LCD ex302.

- La unidad de codificación de la imagen ex312, que incluye el aparato de codificación de imágenes explicado en la presente invención, comprime y codifica los datos de imágenes proporcionados desde la unidad de cámara ex203 por el procedimiento de codificación usado para el aparato de codificación de imágenes según se muestra en las formas de realización mencionadas anteriormente de manera que los transformen en datos de imágenes codificados, y los envía a la unidad de multiplexado/demultiplexado ex308. En ese momento, el teléfono móvil ex115 envía las voces recibidas por la unidad de entrada de voz ex205 durante la captación de imágenes por la unidad de cámara ex203 a la unidad de multiplexado/demultiplexado ex308 como datos de voz digital por medio de la unidad de tratamiento de voz ex305.

- La unidad de multiplexado/demultiplexado ex308 multiplexa los datos de imágenes codificados proporcionados desde la unidad de codificación de la imagen ex312 y los datos de voz proporcionados desde la unidad de tratamiento de voz ex305 por un procedimiento predeterminado, la unidad de circuito de módem ex306 realiza procesamiento de espectro ensanchado de los datos multiplexados resultantes, y la unidad de circuitos de envío/recepción ex301 realiza conversión digital-analógica y transformación de frecuencia sobre el resultado para su emisión por medio de la antena ex201.

- Al igual que para la recepción de datos de un archivo de imágenes en movimiento que está vinculado a una página web o similar en modo de comunicación de datos, la unidad de circuito de módem ex306 realiza procesamiento inverso de espectro ensanchado de los datos recibidos de la estación de base ex110 por medio de la antena ex201, y envía los datos multiplexados resultantes a la unidad de multiplexado/demultiplexado ex308.

Con el fin de decodificar los datos multiplexados recibidos por medio de la antena ex201, la unidad de multiplexado/demultiplexado ex308 demultiplexa los datos multiplexados en un flujo codificado de bits de datos de imágenes y un flujo codificado de bits de datos de voz, y proporciona datos de imágenes codificados a la unidad de comunicación de imágenes ex309 y los datos de voz a la unidad de tratamiento de voz ex305 respectivamente por medio del bus síncrono ex313.

A continuación, la unidad de comunicación de imágenes ex309, que incluye el aparato de decodificación de imágenes explicado en la presente invención, decodifica el flujo codificado de bits de los datos de imágenes por el procedimiento de decodificación emparejado con el procedimiento de codificación según se muestra en las formas de realización mencionadas anteriormente, de manera que se generen los datos de las imágenes en movimiento reproducidas, y proporciona estos datos a la unidad de pantalla ex202 por medio de la unidad de control LCD ex302, y así se muestran los datos de imágenes en movimiento incluidos en un archivo de imágenes en movimiento vinculado, por ejemplo, a una página web. Al mismo tiempo, la unidad de tratamiento de voz ex305 convierte los datos de voz en datos de voz analógica, y proporciona estos datos a la unidad de salida de voz ex208, y así se reproducen los datos de voz incluidos en un archivo de imágenes en movimiento vinculado, por ejemplo, a una página web.

La presente invención no se limita al sistema mencionado anteriormente. Últimamente han sido noticia medios de difusión digital por satélite o de base terrestre, y al menos el aparato de codificación de imágenes o el aparato de decodificación de imágenes en las formas de realización mencionadas anteriormente pueden incorporarse en dicho sistema de difusión digital según se muestra en la fig. 22. Más específicamente, se emite un flujo codificado de bits de información de vídeo desde una estación de difusión ex409 a o en comunicación con un satélite de difusión ex410 por medio de ondas de radio. Al recibirlo, el satélite de difusión ex410 emite ondas de radio para difusión, una antena doméstica ex406 con una función de recepción de difusión desde satélite recibe las ondas de radio, y un aparato como una televisión (receptor) ex401 o un decodificador digital (DD) ex407 decodifica el flujo codificado de bits para reproducción. El aparato de decodificación de imágenes según se muestra en las formas de realización mencionadas anteriormente puede implementarse en el dispositivo de reproducción ex403 para leer un flujo codificado de bits grabado en un medio de almacenamiento ex402 como un CD y DVD que es un medio de grabación y decodificación de los mismos. En este caso, las señales de vídeo reproducidas se visualizan en un monitor ex404. También se concibe para implementar el aparato de decodificación de imágenes en el decodificador digital ex407 conectado a un cable ex405 para una televisión por cable o a la antena ex406 para difusión por satélite y/o de base terrestre de manera que las reproduzca en un monitor ex408 de la televisión. El aparato de decodificación de imágenes puede estar incorporado en la televisión, no en el decodificador digital. O bien, un automóvil ex412 que tiene una antena ex411 puede recibir señales del satélite ex410, la estación de base ex107 o similar para reproducir imágenes en movimiento en un aparato de visualización como un dispositivo de navegación en automóvil ex413 o similar en el automóvil ex412.

Además, el aparato de codificación de imágenes según se muestra en las formas de realización mencionadas anteriormente puede codificar señales de imágenes para grabarlas en un medio de grabación. Como un ejemplo concreto, existe un grabador ex420 como un grabador de DVD para grabar señales de imágenes en un disco DVD ex421 y un disco grabador para grabarlos en un disco duro. También pueden grabarse en una tarjeta SD ex422. Si el grabador ex420 incluye el aparato de decodificación de imágenes según se muestra en las formas de realización mencionadas anteriormente, las señales de imágenes grabadas en el disco DVD ex421 o la tarjeta SD ex422 pueden reproducirse para visualización en el monitor ex408.

Obsérvese que al igual que la estructura del dispositivo de navegación en automóvil ex413, es concebible la estructura sin la unidad de cámara ex203, la unidad de interfaz de la cámara ex303 y la unidad de codificación de la imagen ex312, entre las unidades según se muestra en la fig. 21. Lo mismos se aplica al ordenador ex111, la televisión (receptor) ex401 y otros.

Además, pueden concebirse tres tipos de implementaciones para un terminal como el teléfono móvil ex114 mencionado anteriormente; un terminal de envío/recepción equipado con un codificador y un decodificador, un terminal de envío equipado sólo con un codificador, y un terminal de recepción equipado sólo con un decodificador.

Según se describe anteriormente, es posible usar el procedimiento de codificación de vectores de movimiento o el procedimiento de decodificación de vectores de movimiento según se muestra en las formas de realización anteriores en cualquiera de los dispositivos y sistemas mencionados anteriormente, y así pueden obtenerse los efectos explicados en las formas de realización anteriores.

Aplicabilidad industrial

El procedimiento de codificación de vectores de movimiento y el procedimiento de decodificación de vectores de movimiento según la presente invención son adecuados para su uso en un aparato de codificación de imágenes en movimiento para codificación de imágenes en movimiento, un aparato de decodificación de imágenes en movimiento para decodificación de imágenes en movimiento codificadas, y un sistema que incluya estos aparatos, como un sistema de suministro de contenidos para proporcionar contenidos como, por ejemplo, obras digitales y un sistema de difusión digital.

REIVINDICACIONES

1. Un aparato de decodificación de imágenes que decodifica un bloque actual en una imagen en movimiento, comprendiendo dicho aparato:
 - 5 una unidad de especificación de bloques vecinos accionable para especificar un bloque vecino que está situado en la vecindad del bloque actual y que ya ha sido decodificado,
 - una unidad de valoración accionable para valorar si el bloque vecino ha sido o no decodificado usando un vector de movimiento calculado a partir de un vector de movimiento de otro bloque situado temporalmente después del bloque vecino;
 - 10 una unidad de determinación accionable para determinar un vector de movimiento del bloque vecino que será un vector de movimiento real usado en la realización de compensación de movimiento en el bloque vecino cuando dicha unidad de valoración valora que el bloque vecino ha sido decodificado usando el vector de movimiento calculado a partir de un vector de movimiento de dicho otro bloque;
 - 15 una unidad de predicción de vector de movimiento (706) accionable para obtener un vector de movimiento predictivo del bloque actual usando varios vectores de movimiento de varios bloques vecinos;
 - 20 una unidad de decodificación de vector de movimiento (701) accionable para decodificar un vector de movimiento codificado del bloque actual para obtener un vector de movimiento de diferencia decodificado del bloque actual;
 - una unidad de recuperación de vector de movimiento (706) accionable para recuperar un vector de movimiento del bloque actual sumando el vector de movimiento de diferencia decodificado del bloque actual y el vector de movimiento predictivo;
 - 25 una unidad de compensación de movimiento (705) accionable para generar una imagen con compensación de movimiento usando el vector de movimiento recuperado y una imagen de referencia correspondiente al vector de movimiento recuperado;
 - 30 una unidad de decodificación de imágenes (702) accionable para decodificar un error de predicción entre el bloque actual y la imagen con compensación de movimiento; y
 - 35 una unidad de recuperación de imágenes (708) accionable para obtener una imagen recuperada sumando la imagen con compensación de movimiento y el error de predicción decodificado,
 - en el que dicho otro bloque es un bloque incluido en una imagen diferente a la imagen que incluye el bloque vecino, y el vector de movimiento real del bloque vecino se calcula a partir del vector de movimiento de dicho otro bloque.
 - 40
2. Un procedimiento de decodificación de imágenes para decodificar un bloque actual en una imagen en movimiento, comprendiendo dicho procedimiento:
 - 45 una etapa de especificación de bloque vecino (S200, S400) para especificar un bloque vecino que está situado en la vecindad del bloque actual y que ya ha sido decodificado,
 - una etapa de valoración (S202, S402) para valorar si el bloque vecino ha sido o no decodificado usando un vector de movimiento calculado a partir de un vector de movimiento de otro bloque situado temporalmente después del bloque vecino;
 - 50 una etapa de determinación para determinar un vector de movimiento del bloque vecino que será un vector de movimiento real usado para realizar compensación de movimiento en el bloque vecino, cuando se valora en dicha etapa de valoración (S202, S402) que el bloque vecino ha sido decodificado usando el vector de movimiento calculado a partir de un vector de movimiento de dicho otro bloque;
 - 55 una etapa de predicción de vector de movimiento (S206, S410) para obtener un vector de movimiento predictivo del bloque actual usando varios vectores de movimiento de varios bloques vecinos;
 - una etapa de decodificación de vector de movimiento de diferencia para decodificar un vector de movimiento

codificado del bloque actual para obtener un vector de movimiento de diferencia decodificado del bloque actual;

una etapa de recuperación de vector de movimiento para recuperar un vector de movimiento del bloque actual sumando el vector de movimiento de diferencia decodificado del bloque actual y el vector de movimiento predictivo;

5

una etapa de compensación de movimiento para generar una imagen con compensación de movimiento usando el vector de movimiento recuperado y una imagen de referencia correspondiente al vector de movimiento recuperado;

10 una etapa de decodificación de imágenes para decodificar un error de predicción entre el bloque actual y la imagen con compensación de movimiento; y

una etapa de recuperación de imágenes (S210, S416) para obtener una imagen recuperada sumando la imagen con compensación de movimiento y el error de predicción decodificado,

15 en el que dicho otro bloque es un bloque incluido en una imagen diferente de una imagen que incluye el bloque vecino, y el vector de movimiento real del bloque vecino se calcula a partir del vector de movimiento de dicho otro bloque.

Fig. 1

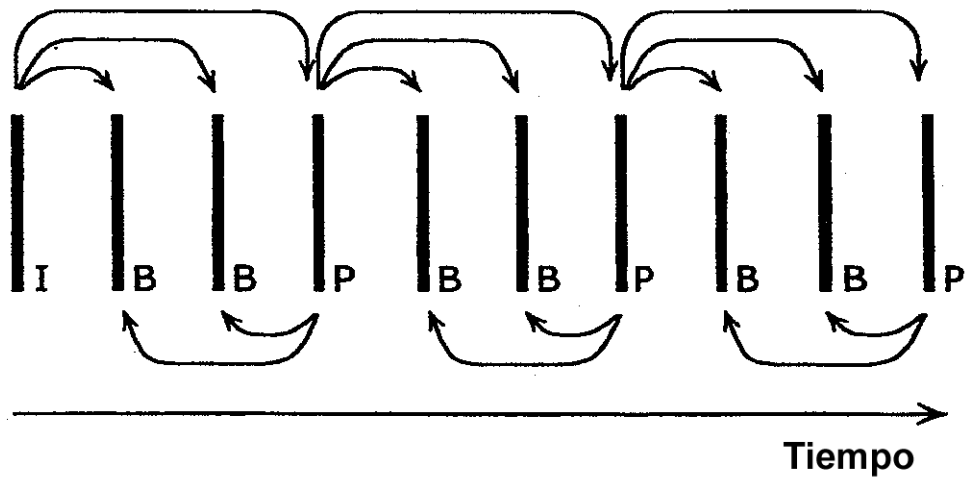


Fig. 2

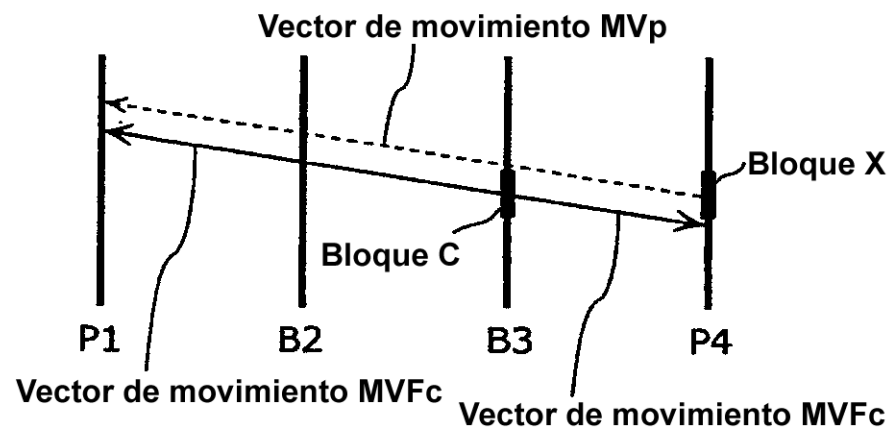


Fig. 3A

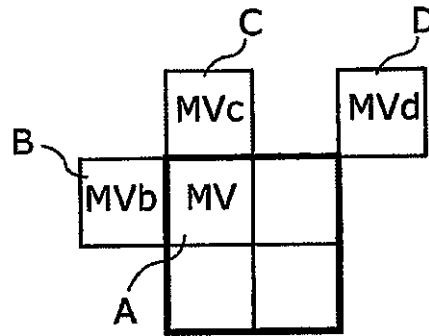


Fig. 3B

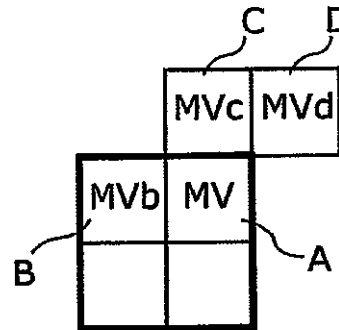


Fig. 3C

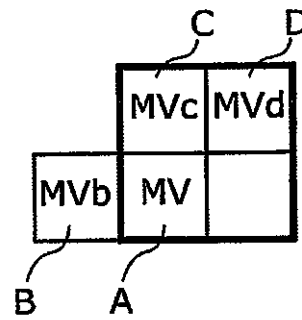


Fig. 3D

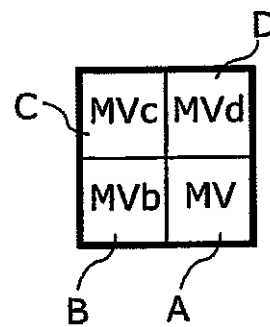


Fig. 4

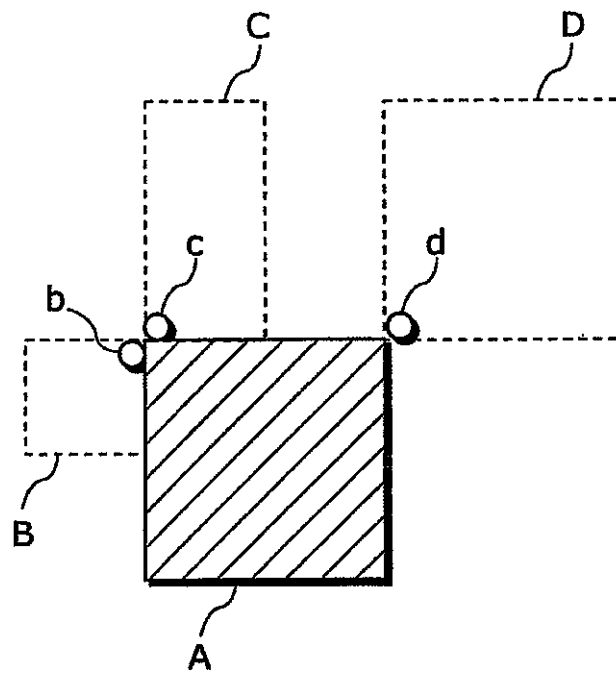
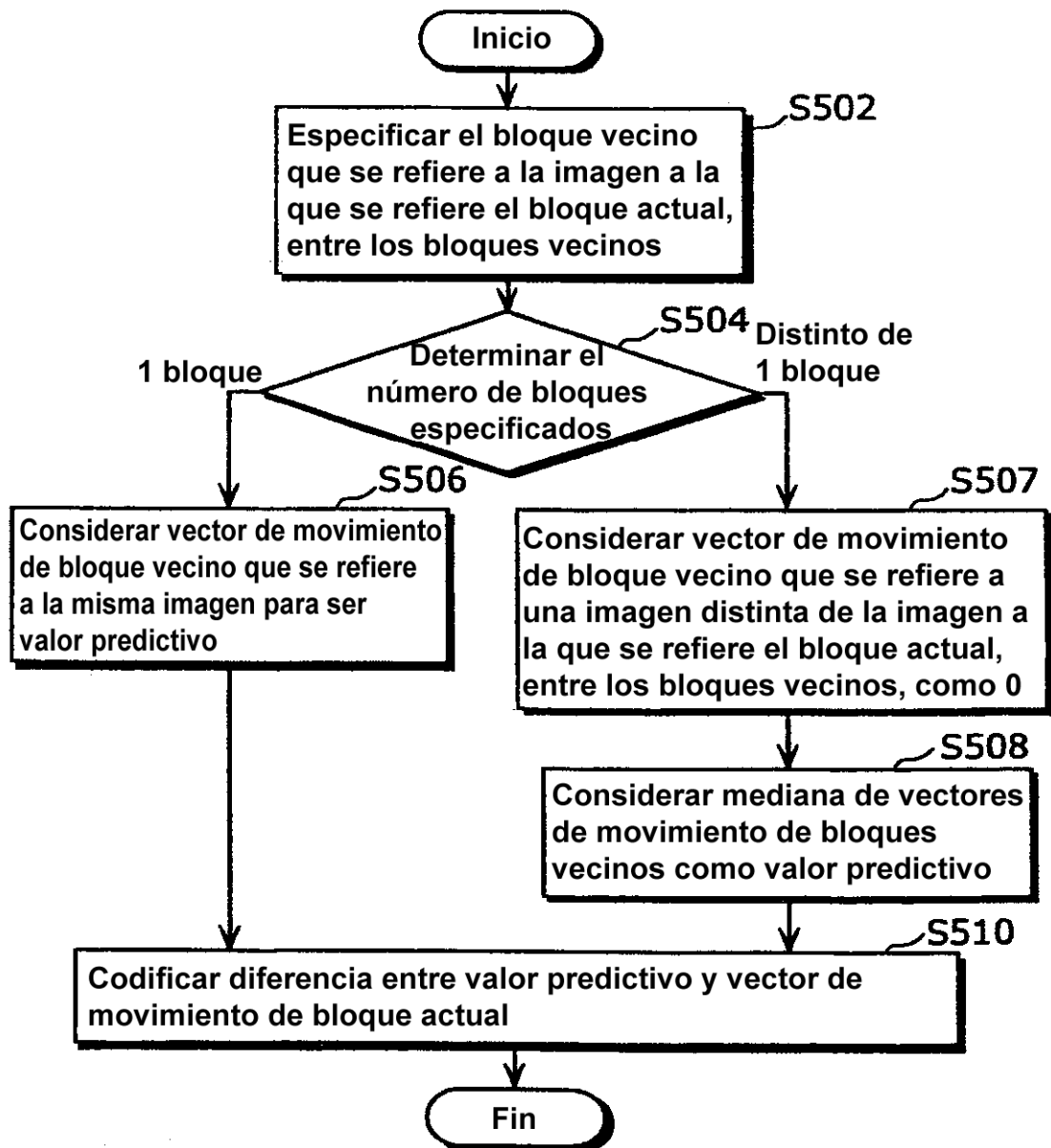
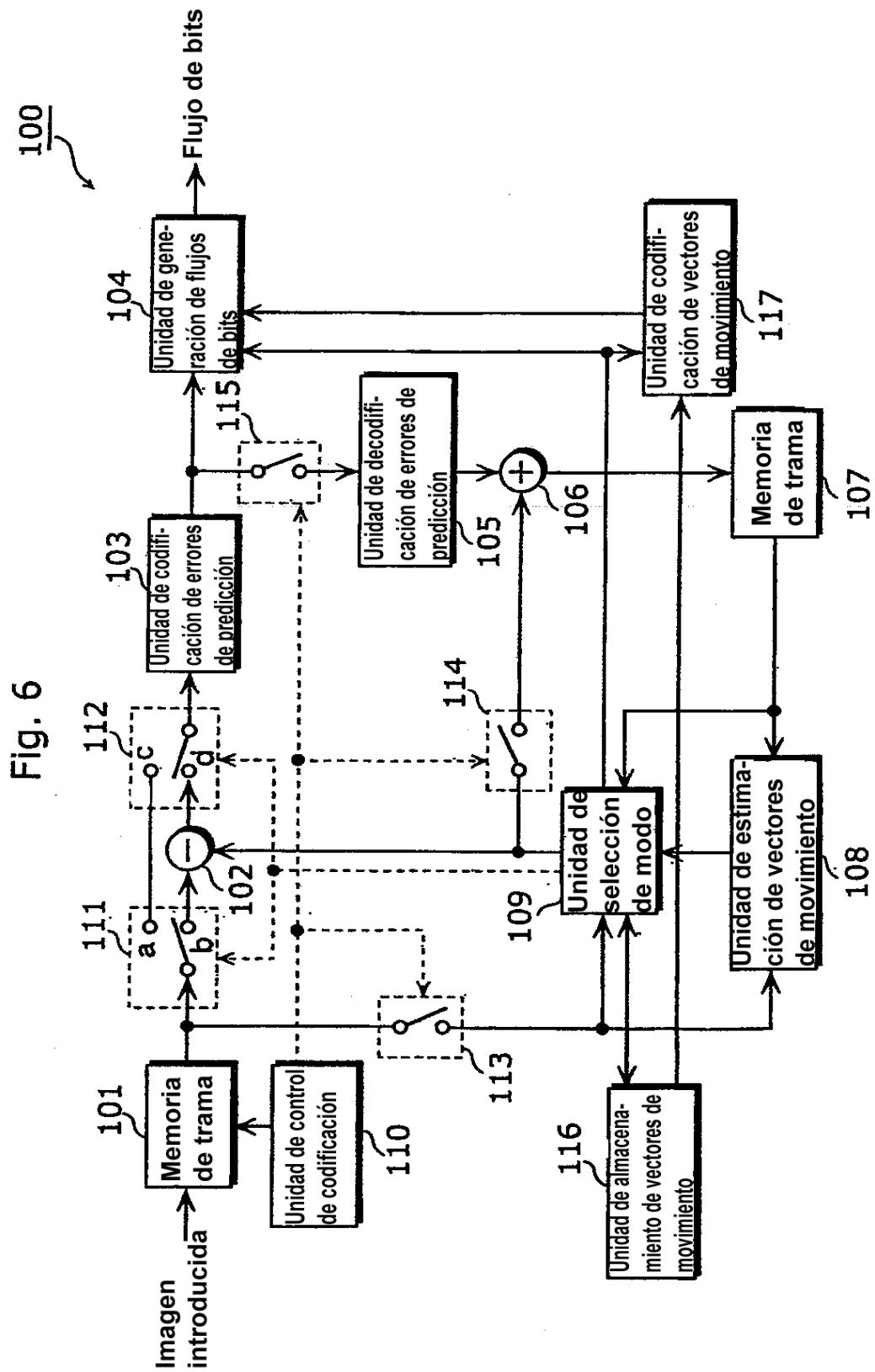
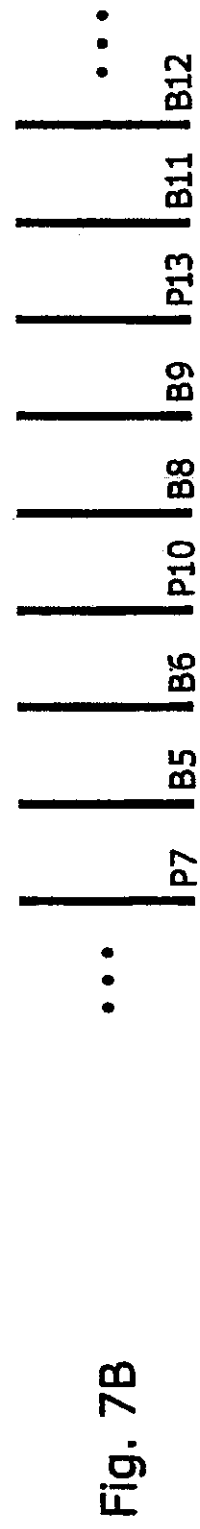
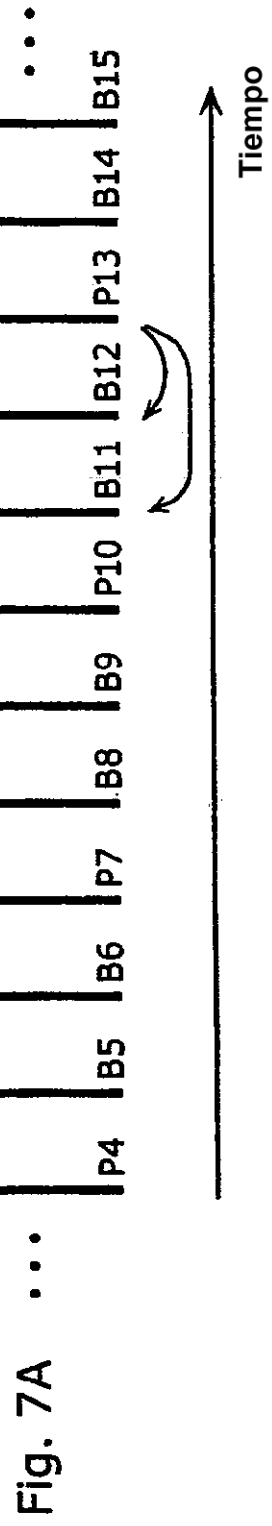


Fig. 5







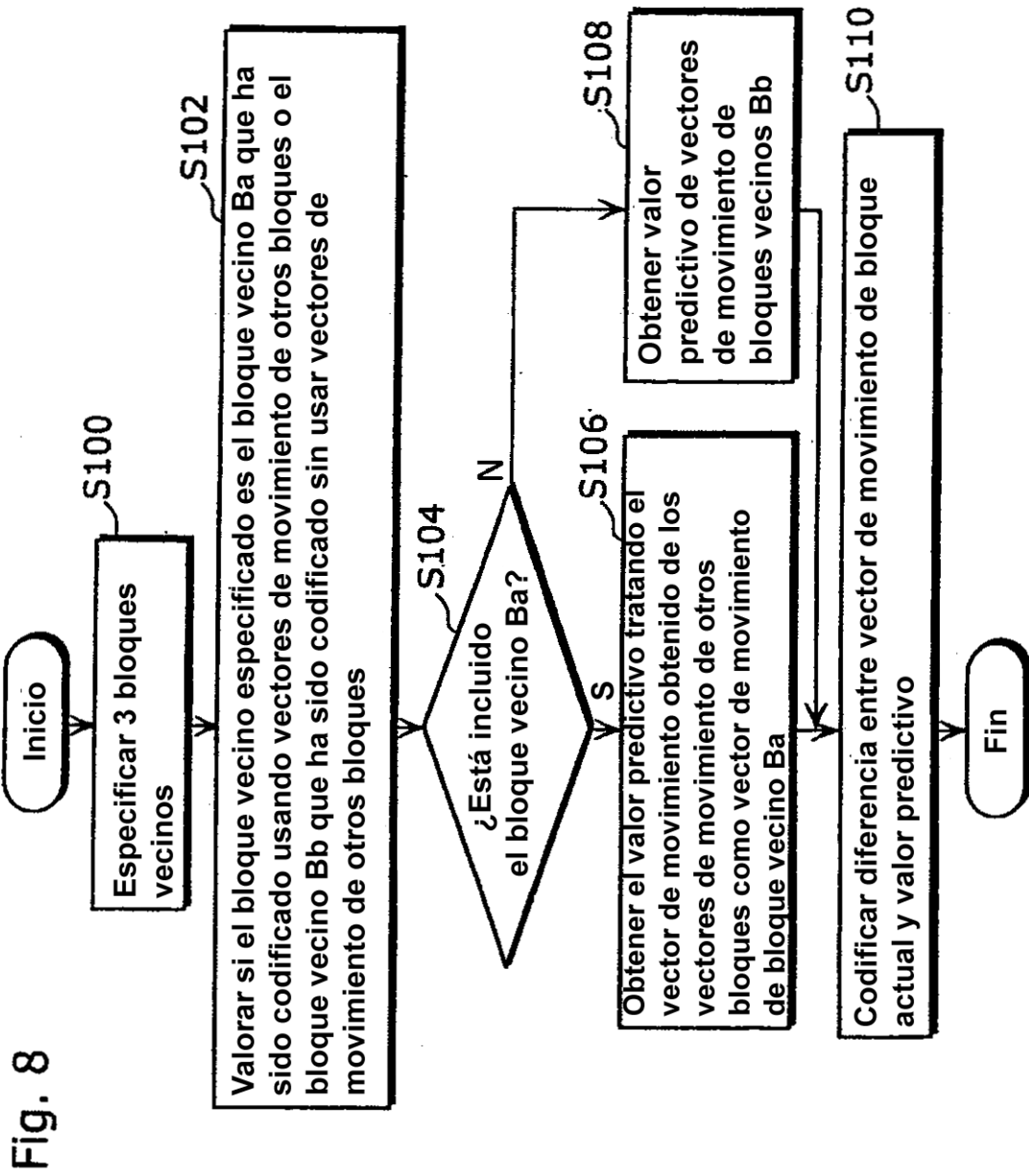


Fig. 9

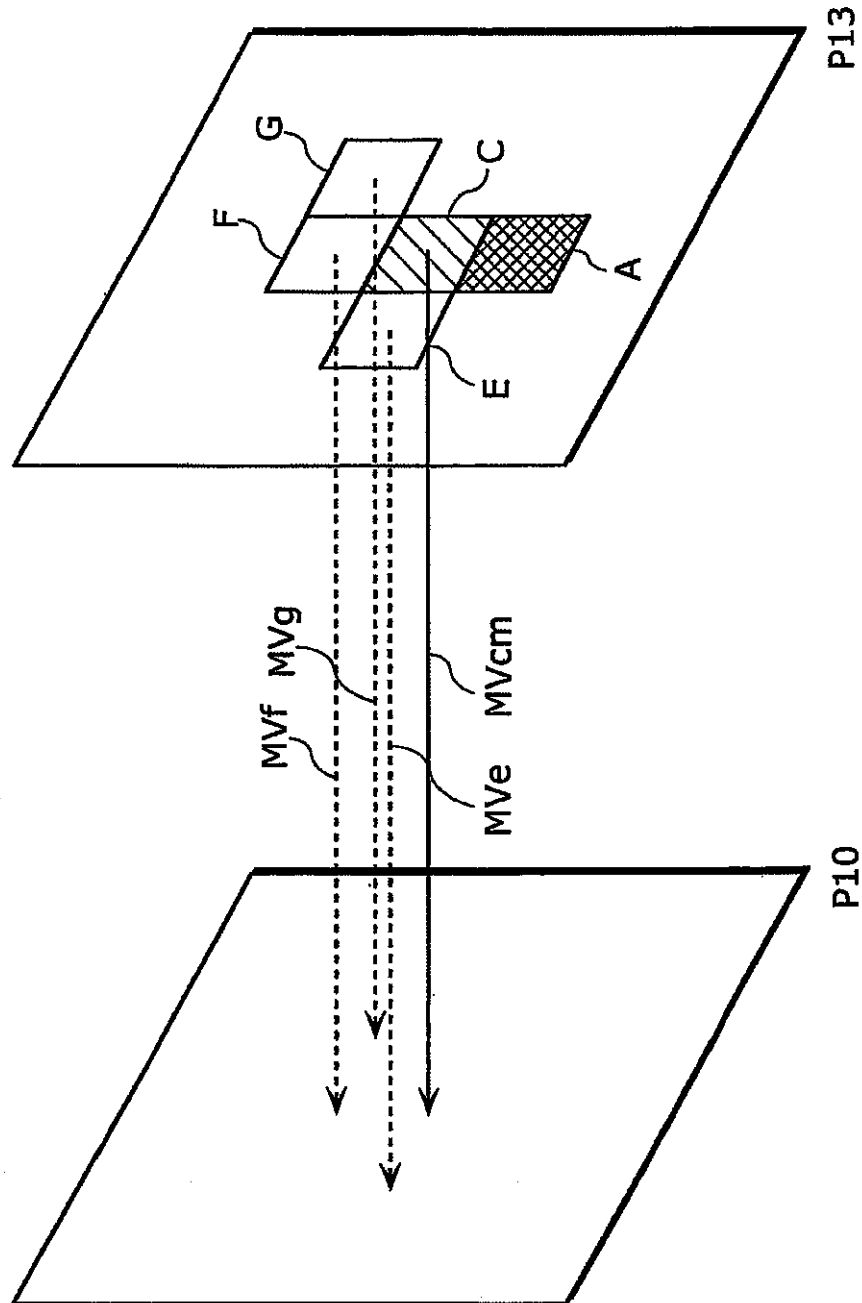


Fig. 10

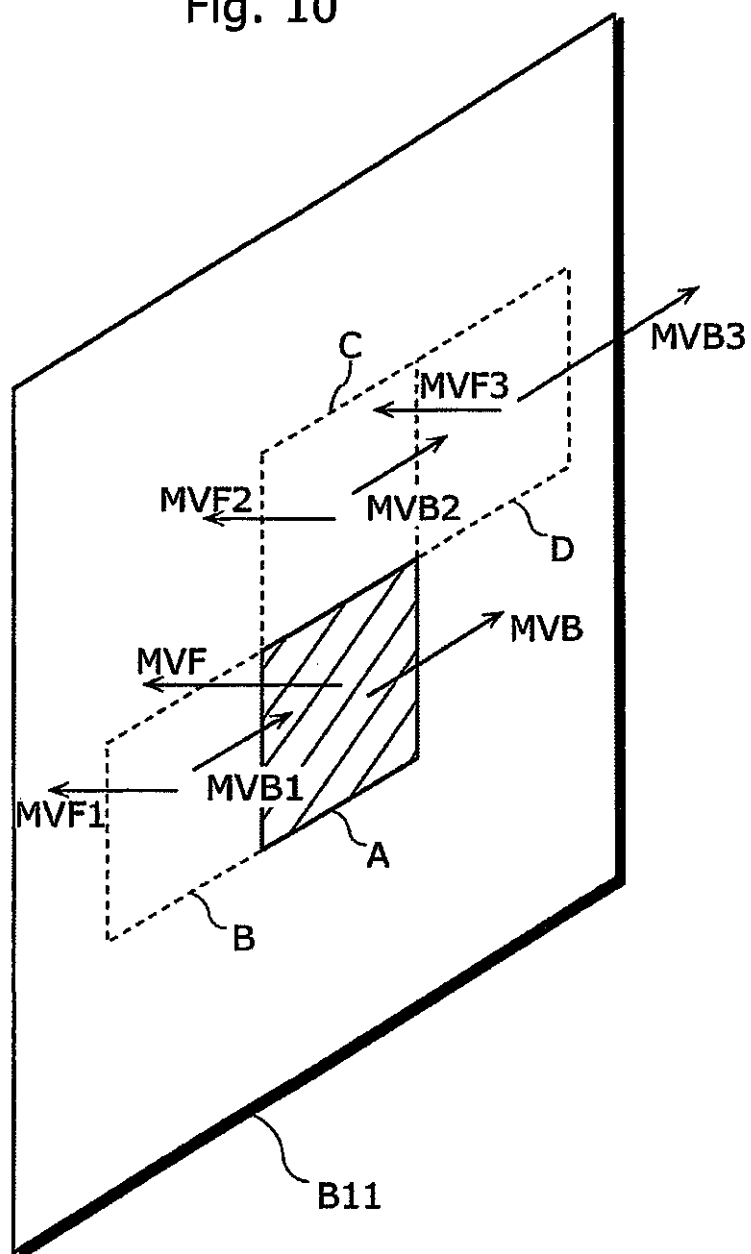


Fig. 11

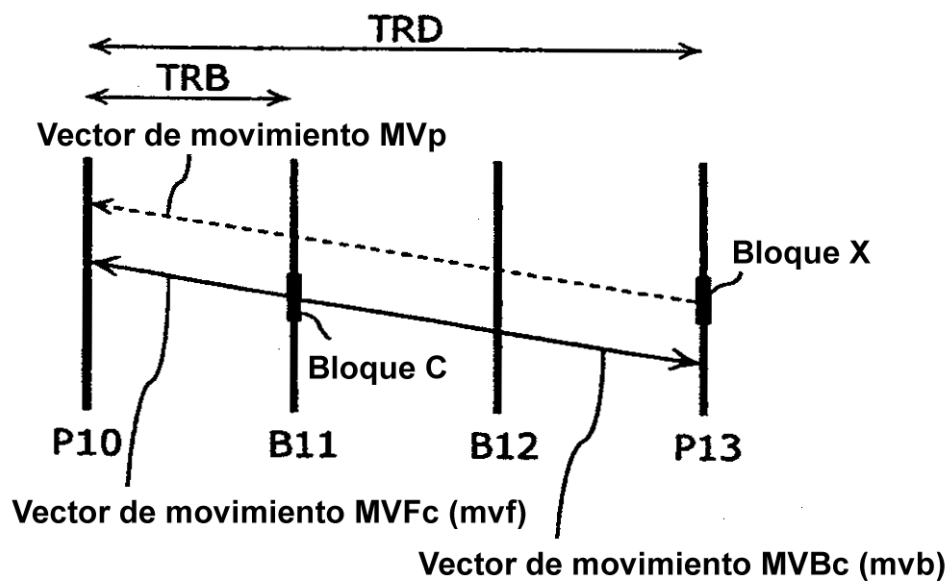


Fig. 12

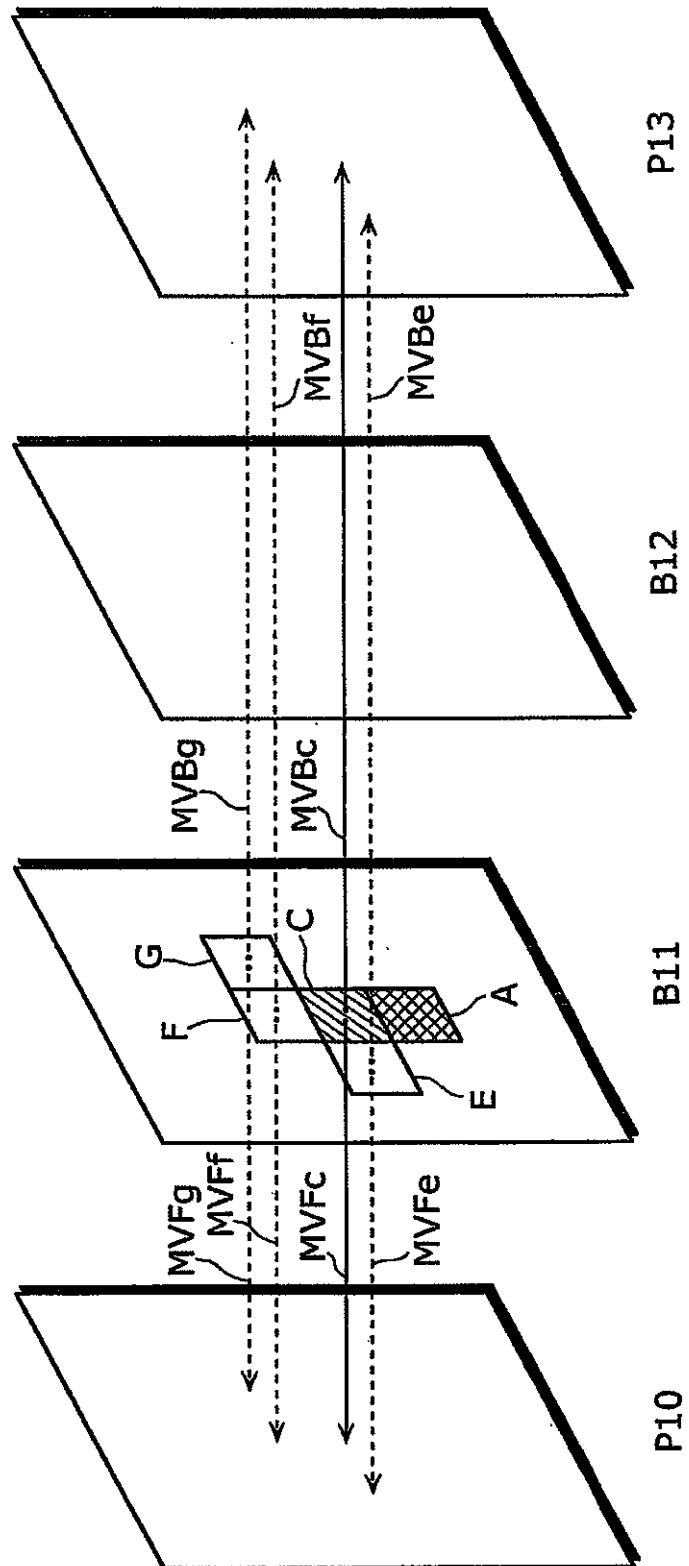
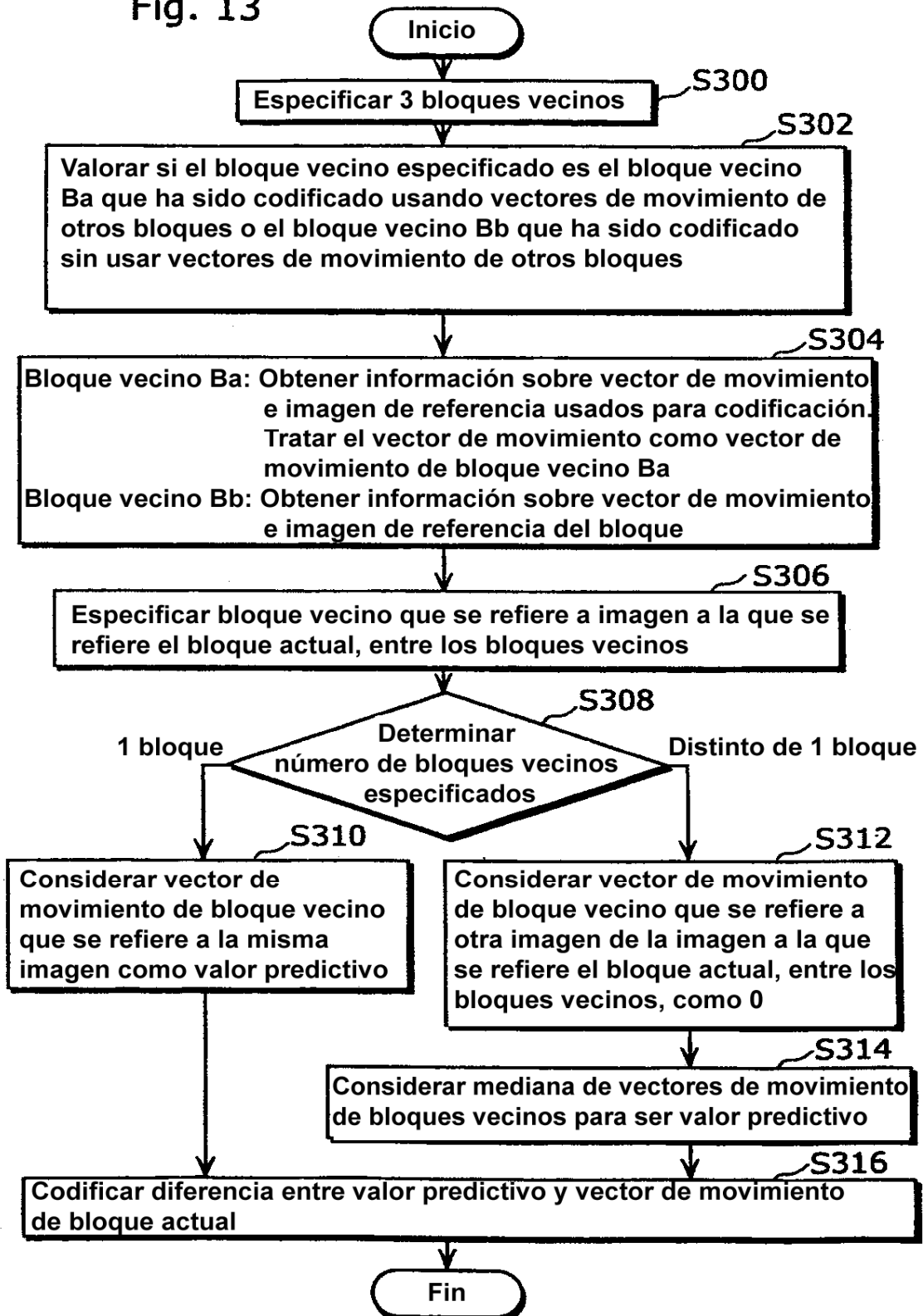
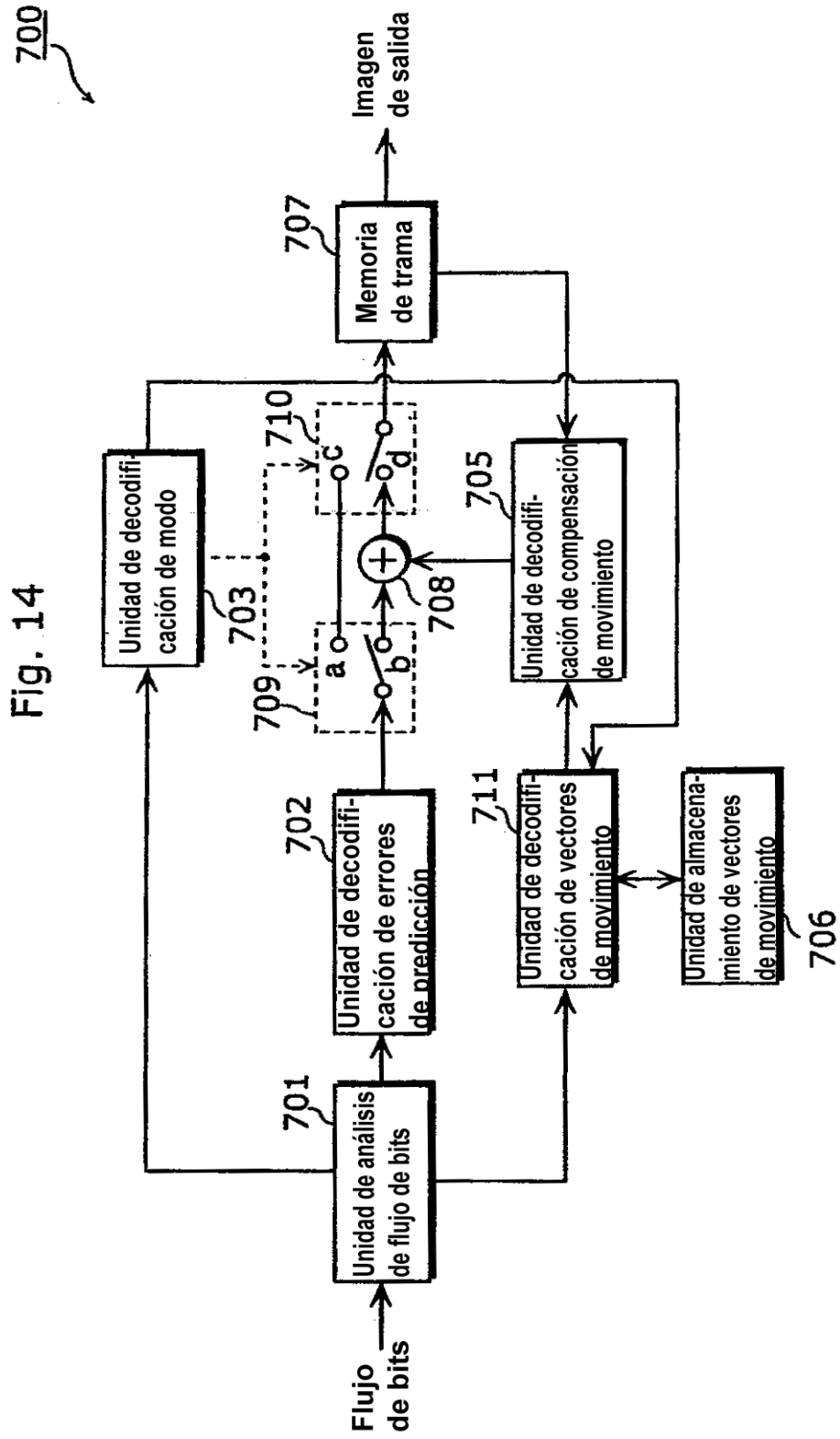
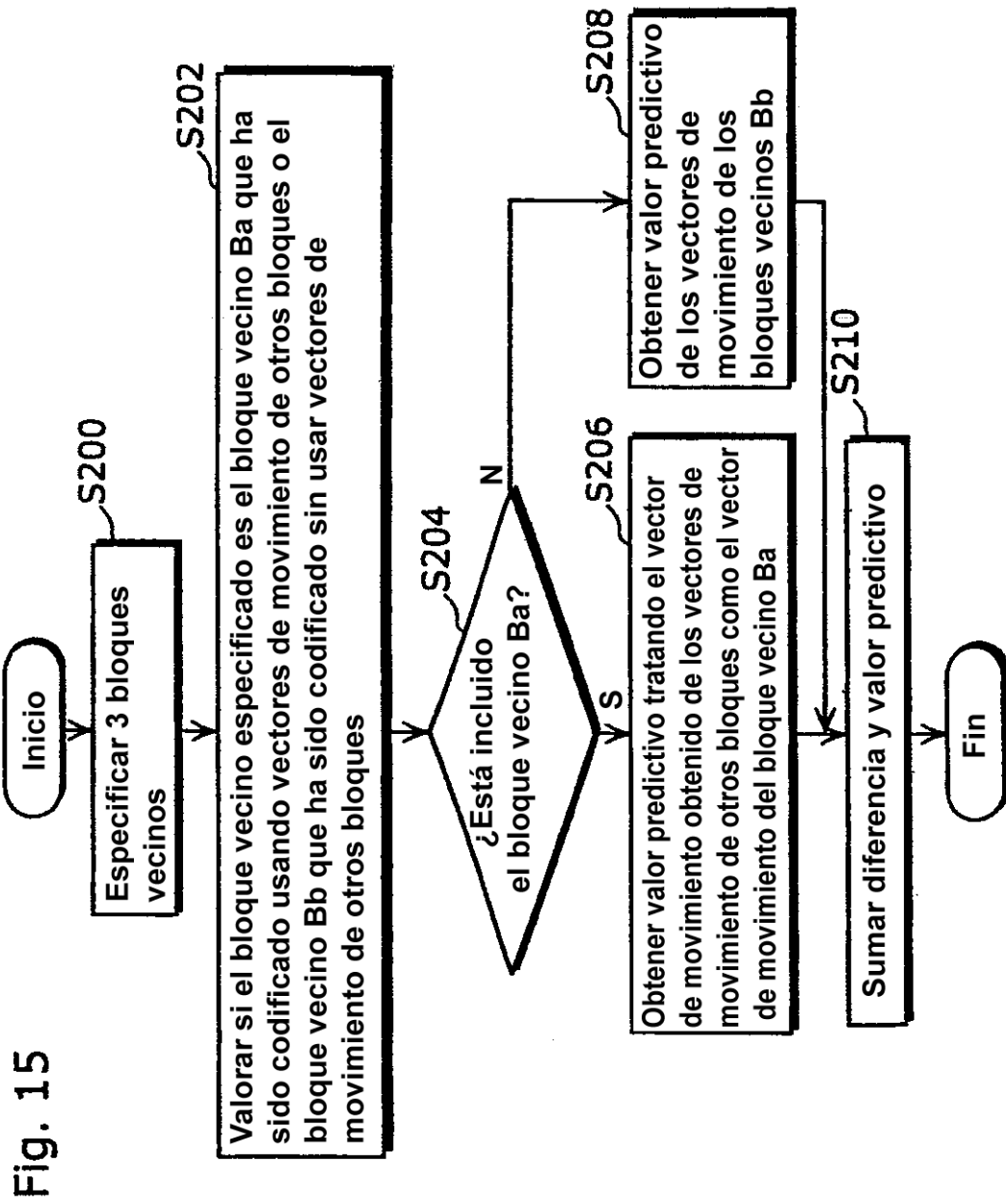


Fig. 13







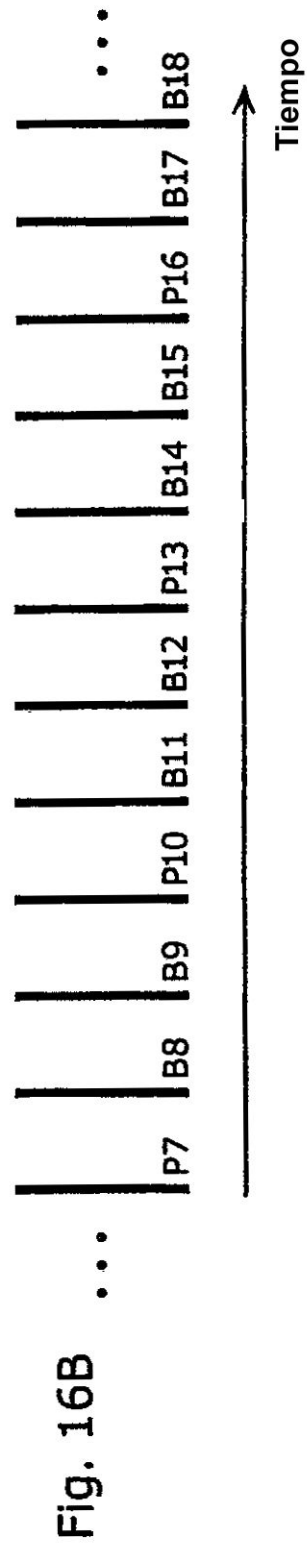
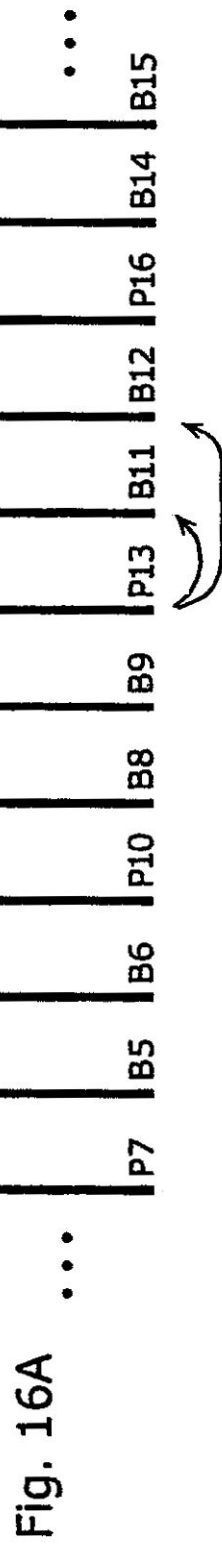
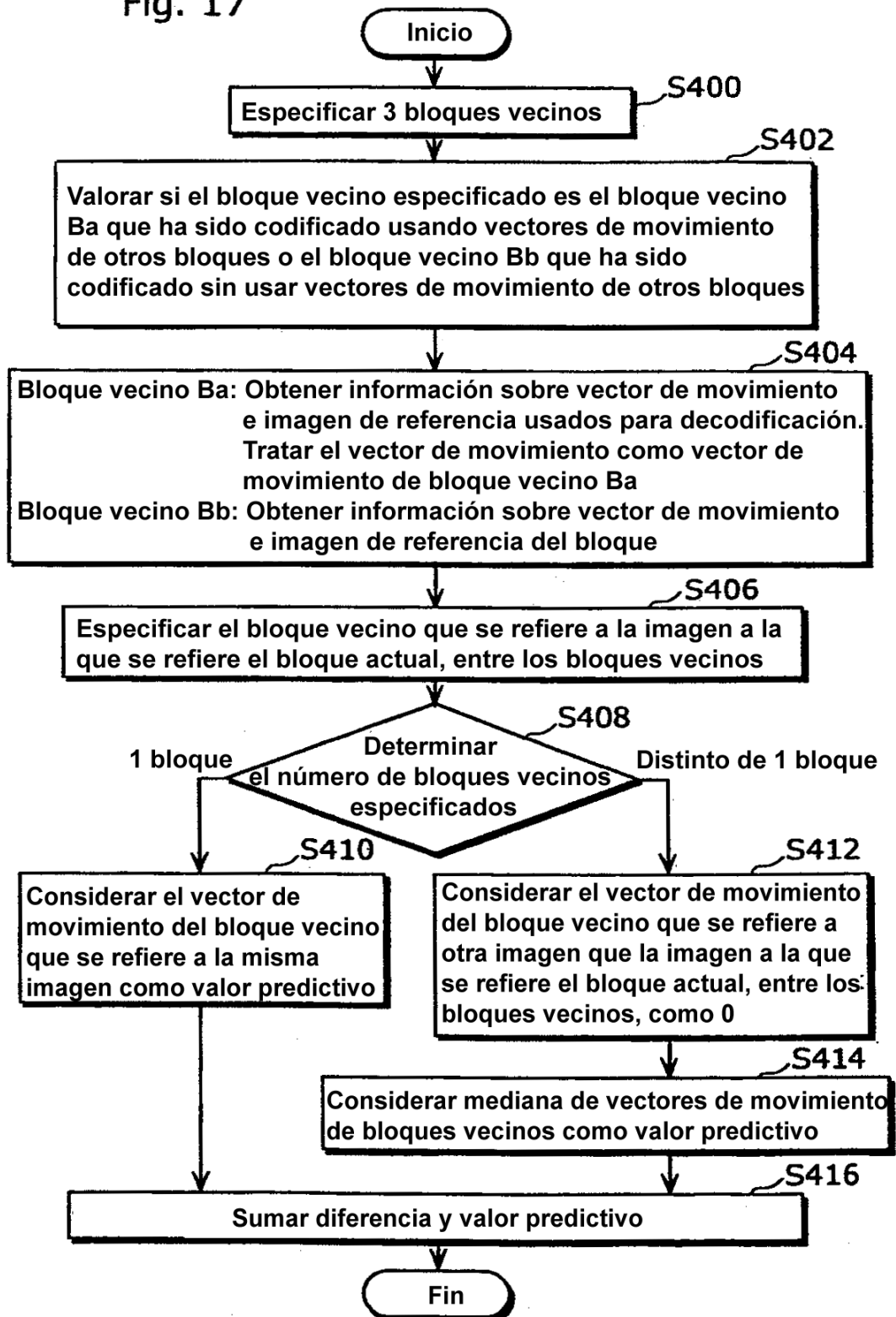
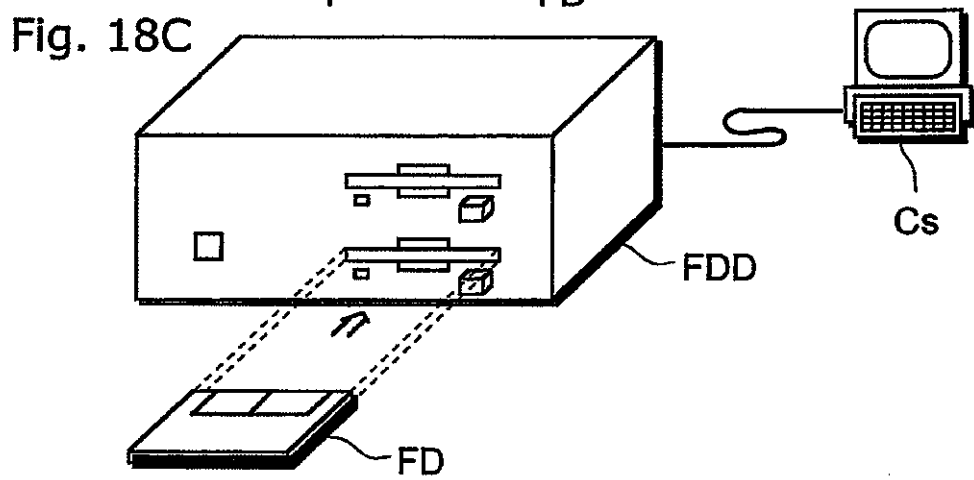
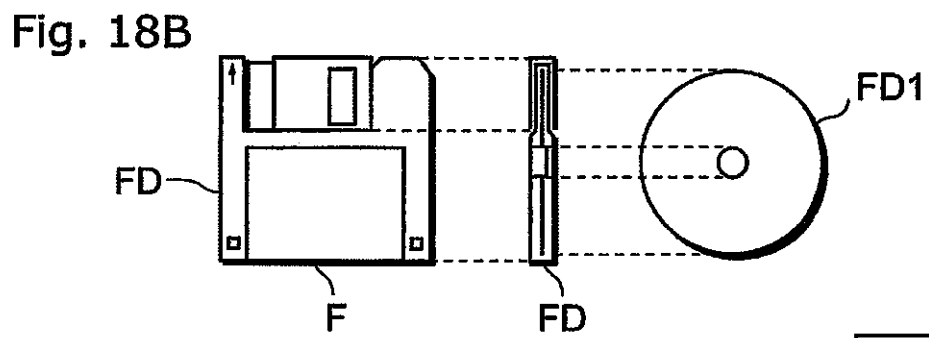
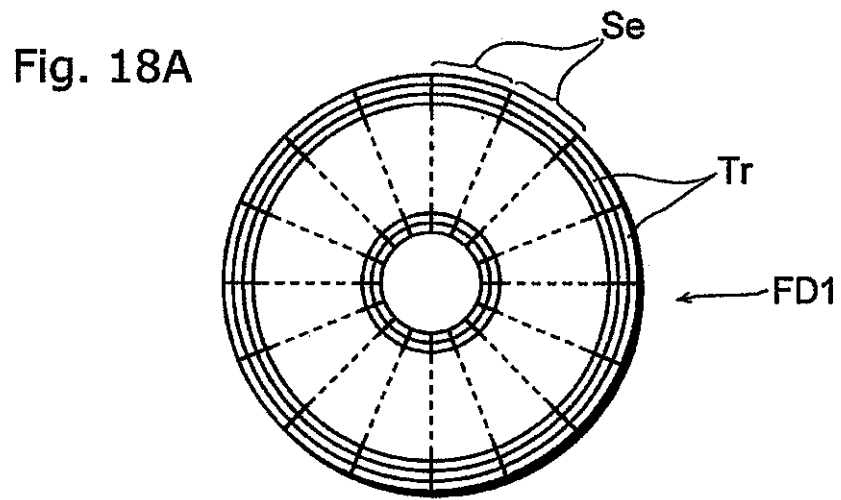


Fig. 17





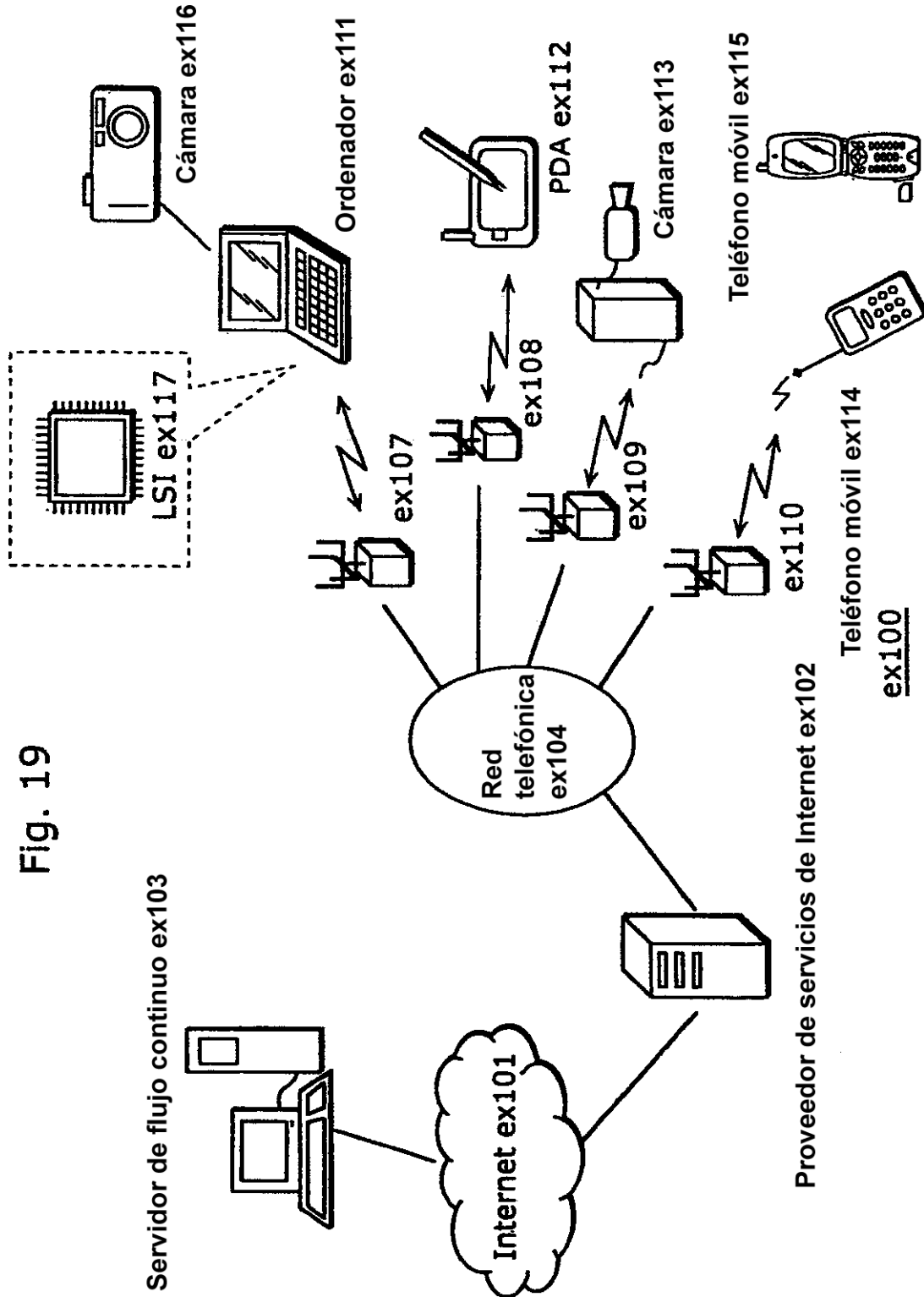
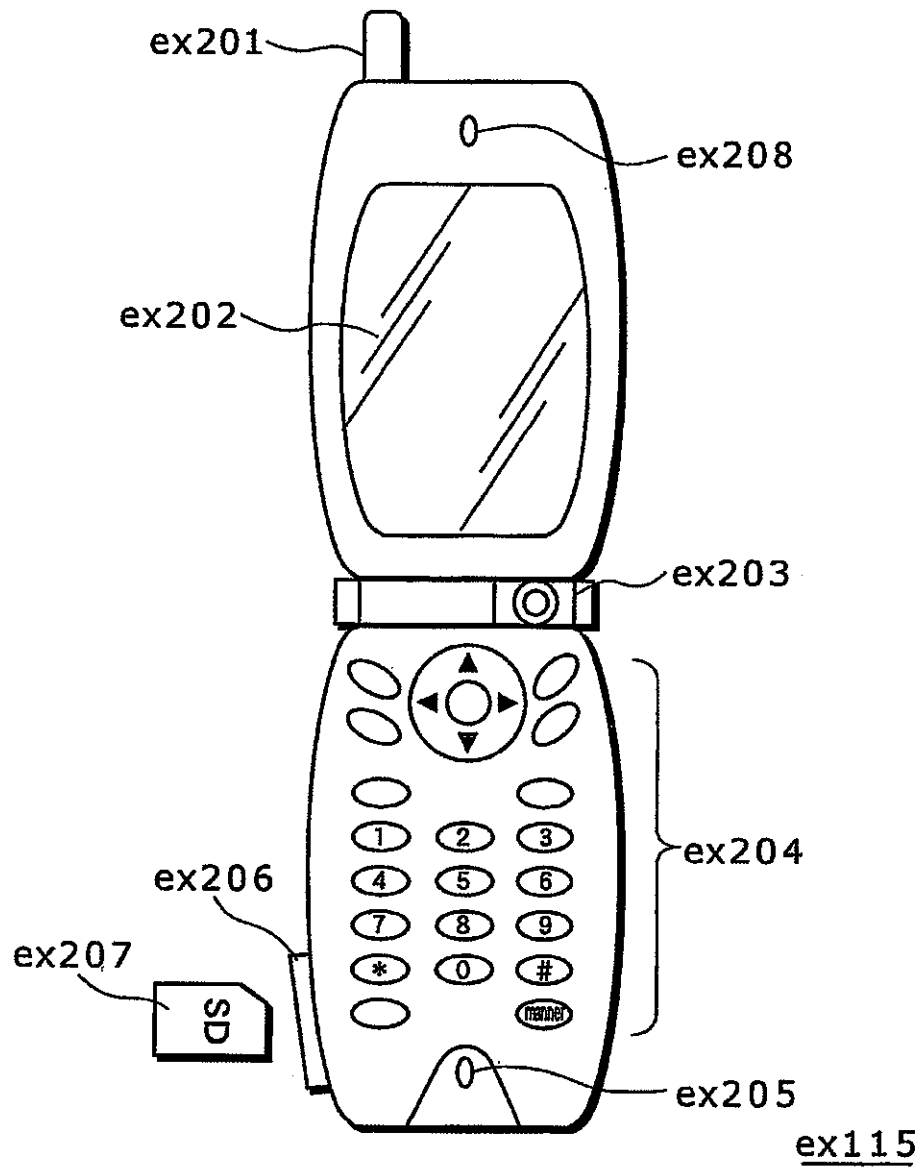


Fig. 20



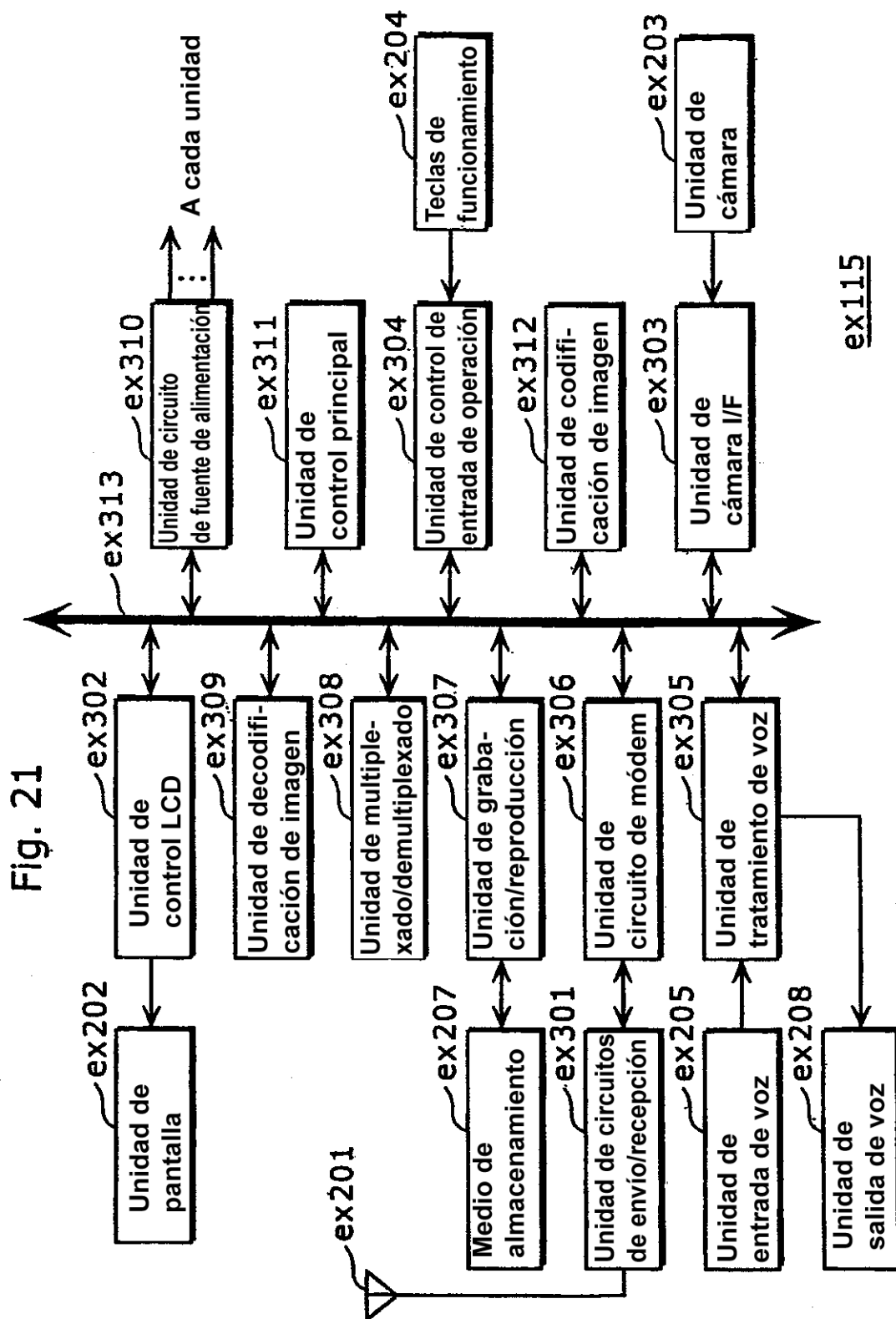


Fig. 22

