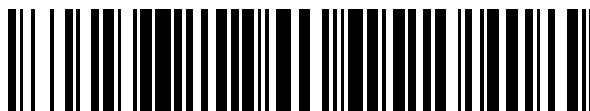


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 380 590**

51 Int. Cl.:
H04N 7/30

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **98921904 .3**

96 Fecha de presentación: **15.05.1998**

97 Número de publicación de la solicitud: **1076998**

97 Fecha de publicación de la solicitud: **21.02.2001**

54 Título: **Procedimiento y aparato para rellenar información de textura entrelazada de macrobloques**

30 Prioridad:
14.03.1998 KR 9808637

45 Fecha de publicación de la mención BOPI:
16.05.2012

45 Fecha de la publicación del folleto de la patente:
16.05.2012

73 Titular/es:
**DAEWOO ELECTRONICS CORPORATION
686, AHYEON-DONG MAPO-GU
SEOUL, KR**

72 Inventor/es:
LEE, Sang, Hoon

74 Agente/Representante:
Sugrañes Moliné, Pedro

ES 2 380 590 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 y aparato para rellenar información de textura entrelazada de macrobloques

5 **Campo técnico de la invención**

La presente invención se refiere a un procedimiento y a un aparato para rellenar información de textura entrelazada de macrobloques; y, más en particular, a un procedimiento y a un aparato que pueden rellenar información de textura acerca de un cuadro anterior para cada macrobloque con el fin de realizar una estimación de movimiento en la técnica de codificación entrelazada.

Técnica anterior

En los sistemas televisados de manera digital tales como videotelefonía, teleconferencia y sistemas de televisión de alta definición, se necesita una gran cantidad de datos digitales para definir cada señal de cuadro de vídeo ya que una señal de línea de vídeo en la señal de cuadro de vídeo comprende una secuencia de datos digitales denominada como valores de píxel. Sin embargo, puesto que el ancho de banda de frecuencia disponible de un canal de transmisión convencional es limitado, para transmitir la gran cantidad de datos digitales a través del mismo es necesario comprimir o reducir el volumen de datos mediante la utilización de varias técnicas de compresión de datos, especialmente en el caso de codificadores de señales de vídeo de baja velocidad binaria tales como sistemas de videotelefonía y de teleconferencia.

Una de estas técnicas para codificar señales de vídeo para un sistema de codificación de baja velocidad binaria es la denominada técnica de codificación de análisis-síntesis orientada a objetos, en la que una imagen de vídeo de entrada se divide en objetos, y tres conjuntos de parámetros que definen los datos de movimiento, de contorno y de píxeles de cada objeto se procesan a través de diferentes canales de codificación.

Un ejemplo del esquema de codificación orientado a objetos es la denominada fase 4 de MPEG (grupo de expertos en imágenes en movimiento, *Moving Picture Express Group*) (MPEG-4), que está diseñada para proporcionar una norma de codificación audiovisual para permitir una interactividad basada en contenido, una eficacia de codificación mejorada y/o una accesibilidad universal en aplicaciones tales como una comunicación de baja velocidad binaria, sistemas multimedia interactivos (por ejemplo, juegos, TV interactiva, etc.) y vigilancia de zona (véase, por ejemplo, el modelo de verificación de vídeo MPEG-4 versión 7.0, Organización Internacional de Normalización, ISO/IEC JTC1/SC29/WG11 MPEG97/N1642, abril de 1997).

Según MPEG-4, una imagen de vídeo de entrada se divide en una pluralidad de planos de objetos de vídeo (VOP), que corresponden a entidades de un flujo de bits a las que un usuario puede acceder y manipular. Un VOP puede denominarse como un objeto y puede representarse mediante un rectángulo delimitador cuya anchura y altura puede ser el múltiplo más pequeño de 16 píxeles (un tamaño de macrobloque) que rodea a cada objeto de manera que el codificador puede procesar la imagen de vídeo de entrada por cada VOP, es decir, por cada objeto.

Cada VOP en MPEG-4 se describe mediante tres conjuntos de parámetros que definen la información de forma, la información de movimiento y la información de textura, es decir, información de color de los mismos, donde la información de forma representada mediante, por ejemplo, una máscara binaria corresponde al contorno de cada dicho VOP, es decir, el borde de un objeto; la información de movimiento representa la correlación espacial entre un cuadro actual y un cuadro anterior correspondiente; y la información de textura consiste en datos de luminancia y de crominancia.

Por otro lado, puesto que la información de textura para dos imágenes de vídeo de entrada recibidas secuencialmente tiene naturalmente redundancia de tiempo, es deseable eliminar la redundancia de tiempo en la misma utilizando una estimación de movimiento y una técnica de compensación con el fin de codificar de manera eficaz la información de textura en MPEG-4.

Una técnica de relleno progresivo de imágenes, es decir, una técnica de relleno repetitivo convencional, se aplica por cada cuadro en el VOP antes de la estimación y compensación de movimiento. En principio, la técnica de relleno repetitivo llena el área transparente fuera del objeto del VOP repitiendo píxeles frontera del objeto, donde los píxeles frontera están ubicados en el contorno del objeto. Es preferible llevar a cabo la técnica de relleno repetitivo con respecto a la información de forma reconstruida, que se genera descodificando la información de forma codificada en el orden inverso a lo largo del esquema de codificación. Cada píxel frontera se repite hacia el exterior del objeto. Si los píxeles transparentes del área transparente fuera del objeto pueden llenarse mediante la repetición de más de un píxel frontera, es preferible tomar el promedio de los valores repetidos como un valor de relleno. Este proceso de relleno progresivo se divide generalmente en tres etapas, a saber, un relleno repetitivo horizontal, un relleno repetitivo vertical y un relleno exterior (véase el modelo de verificación de vídeo MPEG-4 versión 7.0).

Aunque la técnica de relleno progresivo se utiliza para llevar a cabo la estimación y compensación de movimiento en función de cada cuadro generado cada 1/30 segundos tal y como se ha descrito anteriormente, se necesita una

técnica de relleno entrelazado para llevar a cabo la estimación y compensación de movimiento por cada campo, donde dos campos, a saber, un campo superior y un campo inferior, se combinan para reconstruirse como un cuadro, es decir, una imagen de textura entrelazada. La técnica de codificación entrelazada para llevar a cabo la estimación y compensación de movimiento por cada campo puede utilizarse de manera adecuada para codificar de manera precisa la información de textura entrelazada para los movimientos rápidos, tales como en un deporte, en una carrera de caballos y en una carrera de coches, o con una correlación de campo poco frecuente, es decir, con una correlación temporal poco frecuente entre el campo superior y el campo inferior.

Sin embargo, si el relleno para cada bloque de campo se realiza de manera independiente sin considerar su correlación de campo basada en la información de textura entrelazada tal y como se describe en la técnica de relleno progresivo, puede haber algún píxel que no se modifique en función de solamente los píxeles de objeto del objeto, sino que se modificará mediante relleno cuando se tenga en cuenta su correlación de campo, necesitándose dos campos consecutivos, es decir, el campo superior y el campo inferior, incluidos en la información de textura entrelazada.

El documento EP 577 365 A1 describe un procedimiento de codificación y de descodificación para señales de imágenes que cambian de manera adaptativa desde un modo de inhibición de codificación predictiva entre cuadros hasta un modo de inhibición de codificación predictiva entre campos sobre macrobloques completos de datos de entrelazado escaneados de cada porción. El procedimiento de codificación de señales de imágenes para datos de imagen escaneados y entrelazados requiere recibir los datos entrelazados y transformarlos utilizando una transformación DCT basada en campos o basada en cuadros seleccionada de manera adaptativa. Los datos escaneados se dividen en macrobloques y cada bloque se somete a la transformación. Se almacena un cuadro anterior de datos escaneados y se proporciona compensación para el movimiento entre los datos de imagen entrelazados recibidos y el cuadro de datos anterior con el fin de generar datos de imagen predictivos. Estos datos se suprimen de los datos recibidos para proporcionar datos que van a transformarse.

Divulgación de la invención

Por lo tanto, un objeto de la invención es proporcionar un procedimiento y un aparato para rellenar la información de textura entrelazada con su correlación de campo por cada macrobloque de textura para llevar a cabo una estimación y compensación de movimiento.

Según la invención se proporciona un procedimiento y un aparato según las reivindicaciones independientes. Realizaciones preferidas de la misma se definen en las reivindicaciones dependientes.

Breve descripción de los dibujos

Los objetos y características anteriores, así como otros adicionales, de la presente invención resultarán evidentes a partir de la siguiente descripción de realizaciones preferidas proporcionadas junto con los dibujos adjuntos, en los que:

la Fig. 1 muestra un diagrama de bloques esquemático de un aparato convencional para codificar información de textura entrelazada de un objeto en una señal de vídeo;

la Fig. 2 presenta un diagrama de flujo para ilustrar el funcionamiento del circuito de procesamiento de cuadro anterior mostrado en la Fig. 1 según la presente invención;

las Fig. 3A y 3B describen un macrobloque frontera a modo de ejemplo y un bloque de campo frontera superior y otro inferior para el macrobloque frontera, respectivamente;

las Fig. 3C a 3E representan un procedimiento de relleno de los bloques de campo frontera superior e inferior de manera secuencial según la presente invención; y

la Fig. 4 ilustra una pluralidad de bloques adyacentes no definidos para un VOP a modo de ejemplo y las direcciones de relleno para cada bloque adyacente no definido.

Modo de llevar a cabo la invención

Haciendo referencia a la Fig. 1, se muestra un diagrama de bloques esquemático de un aparato para codificar información de textura entrelazada en un cuadro actual, donde la información de textura entrelazada puede denominarse como un VOP actual en MPEG-4. La información de textura entrelazada se divide en una pluralidad de macrobloques de textura que van a aplicarse a un circuito de división en cuadros 102, donde cada macrobloque de textura tiene $M \times N$ píxeles de textura, siendo M y N enteros pares positivos comprendidos normalmente entre 4 y 16.

El circuito de división en cuadros 102 divide cada macrobloque de textura en un bloque de campo superior y un bloque de campo inferior, donde el bloque de campo superior que tiene $M/2 \times N$ píxeles de textura contiene cada fila

impar de dicho macrobloque de textura, y el bloque de campo inferior que tiene los otros $M/2 \times N$ píxeles de textura contiene cada fila par de dicho macrobloque de textura. El bloque de campo superior y el bloque de campo inferior de dicho macrobloque de textura se proporcionan secuencialmente como un bloque de campo superior actual y un bloque de campo inferior actual, respectivamente, a un restador 104 y a un estimador de movimiento 116.

5

Casi al mismo tiempo se extrae información de textura entrelazada anterior desde un circuito de procesamiento de cuadro anterior 114 y se proporciona al estimador de movimiento 116 y a un compensador de movimiento 118, donde la información de textura entrelazada anterior de un cuadro anterior tiene píxeles de textura anteriores y precede a la información de textura entrelazada del cuadro actual en un periodo de cuadro. El cuadro anterior también se divide en una pluralidad de regiones de búsqueda y cada región de búsqueda se divide en una región de búsqueda superior y una región de búsqueda inferior, donde la región de búsqueda superior que tiene P ($M/2 \times N$) píxeles de textura anteriores contiene cada fila impar de cada dicha región de búsqueda, y la región de búsqueda inferior que tiene los otros P ($M/2 \times N$) píxeles de textura anteriores contiene cada fila par de cada dicha región de búsqueda, siendo P un entero positivo, normalmente 2.

10

15

El estimador de movimiento 116 determina vectores de movimiento para el VOP actual por cada campo. En primer lugar, el estimador de movimiento 116 detecta un macrobloque de textura anterior en el cuadro anterior para cada bloque de campo superior o inferior actual, donde el macrobloque de textura anterior está ubicado en la misma posición que cada dicho bloque de campo superior o inferior actual. Después, el macrobloque de textura anterior se divide en un bloque de campo superior anterior y en un bloque de campo inferior anterior; por último se calculan errores entre dicho bloque de campo superior o inferior actual y dos bloques de campo superior e inferior anteriores. Puesto que las regiones de búsqueda superior e inferior tienen una pluralidad de bloques de campo superior candidatos y de bloques de campo inferior candidatos incluyendo los bloques de campo superior anteriores y los bloques de campo inferior anteriores, respectivamente, cada dicho bloque de campo superior o inferior actual está desplazado por cada píxel dentro de las regiones de búsqueda superior e inferior según cada bloque de campo superior candidato y según cada bloque de campo inferior candidato, respectivamente; y, en cada desplazamiento, se comparan entre sí dos errores entre cada dicho bloque de campo superior o inferior actual y cada dicho bloque de campo superior candidato y cada dicho bloque de campo inferior candidato.

20

25

30

Tal y como se ha descrito anteriormente, el estimador de movimiento 116 realiza una estimación de movimiento de cada dicho bloque de campo superior o inferior actual con respecto a sus bloques de campo superior e inferior anteriores y selecciona, como un bloque de campo candidato óptimo o un bloque de campo muy similar, un bloque de campo superior o inferior candidato que proporcione un error mínimo. Los datos de salida del estimador de movimiento 116 son un vector de movimiento y una bandera de indicación de campo que se proporcionan al compensador de movimiento 118 y a un circuito de codificación de longitud variable (VLC) 108, donde el vector de movimiento denota un desplazamiento entre cada dicho bloque de campo superior o inferior actual y el bloque de campo candidato óptimo, y la bandera de indicación de campo representa si el bloque de campo candidato óptimo pertenece o no a la región de búsqueda superior.

35

40

El compensador de movimiento 118 lleva a cabo una compensación de movimiento desplazando el bloque de campo candidato óptimo, mediante el vector de movimiento, hacia dicho bloque de campo superior o inferior actual dependiendo de la bandera de indicación de campo para proporcionar de esa manera al restador 104 y a un sumador 112 un bloque de campo candidato óptimo compensado por movimiento como un bloque de campo superior o inferior estimado para cada dicho bloque de campo superior o inferior actual.

45

El restador 104 obtiene un bloque de campo de error suprimiendo el bloque de campo superior o inferior estimado de cada dicho bloque de campo superior o inferior actual por cada píxel correspondiente para proporcionar el bloque de campo de error a un circuito de codificación de textura 106.

50

En el circuito de codificación de textura 106, el bloque de campo de error puede someterse a una transformada ortogonal para eliminar la redundancia espacial del mismo, cuantificándose posteriormente coeficientes de transformada para proporcionar de ese modo los coeficientes de transformada cuantificados al circuito VLC 108 y a un circuito de reconstrucción de textura 110. Puesto que una transformada ortogonal convencional, tal como una transformada del coseno discreta (DCT), se lleva a cabo por cada bloque DCT, presentando normalmente cada bloque DCT 8×8 , es decir, $M/2 \times N/2$ píxeles de textura, el bloque de campo de error que tiene 8×16 píxeles de textura de error puede dividirse preferiblemente en dos bloques DCT en el circuito de codificación de textura 106. Si fuera necesario, antes de realizar la DCT, cada bloque de campo de error puede rellenarse mediante la DCT en función de la información de forma o de la información de forma reconstruida de cada VOP con el fin de reducir las componentes de frecuencia superiores que pueden generarse en el procesamiento DCT. Por ejemplo, un valor predeterminado, por ejemplo '0', puede asignarse a los píxeles de textura de error en el exterior del contorno de cada VOP.

55

60

El circuito VLC 108 lleva a cabo una codificación estadística en los coeficientes de transformada cuantificados introducidos desde el circuito de codificación de textura 106 y en la bandera de indicación de campo y el vector de movimiento, para cada dicho bloque de campo superior o inferior actual, introducidos desde el estimador de movimiento 116 utilizando, por ejemplo, una técnica de codificación de longitud variable convencional, para

65

proporcionar de ese modo datos codificados de manera estadística a un transmisor (no mostrado) para la transmisión de los mismos.

Al mismo tiempo, el circuito de reconstrucción de textura 110 lleva a cabo una cuantificación inversa y una transformada inversa en los coeficientes de transformada cuantificados para proporcionar al sumador 112 un bloque de campo de error reconstruido que corresponde al bloque de campo de error. El sumador 112 combina el bloque de campo de error reconstruido del circuito de reconstrucción de textura 110 y el bloque de campo superior o inferior estimado del compensador de movimiento 118 por cada píxel para proporcionar de ese modo al circuito de procesamiento de cuadro anterior 114 un resultado combinado como un bloque de campo superior o inferior reconstruido para cada dicho bloque de campo superior o inferior actual.

El circuito de procesamiento de cuadro anterior 114 modifica secuencialmente el bloque de campo superior o inferior reconstruido basándose en la información de forma o en la información de forma reconstruida para el VOP actual, para proporcionar de ese modo al estimador de movimiento 116 y al compensador de movimiento 118 todos los bloques de campo superior e inferior modificados como otra información de textura entrelazada anterior para un VOP actual posterior.

Con referencia a la Fig. 2, se muestra un diagrama de flujo que ilustra un aspecto importante de la invención relacionado con el funcionamiento del circuito de procesamiento de cuadro anterior 114 mostrado en la Fig. 1, es decir, que explica un procedimiento de relleno para la información de textura en un cuadro anterior por cada macrobloque.

En la etapa S201 se recibe secuencialmente el bloque de campo superior o inferior reconstruido para el bloque de campo superior o inferior actual y, en la etapa S203 se eliminan los píxeles exteriores del bloque de campo superior o inferior reconstruidos en función de la información de forma del VOP actual, donde los píxeles exteriores están ubicados fuera del contorno del VOP actual. La información de forma reconstruida puede utilizarse en representación de la información de forma, donde la información de forma se codifica según un esquema de codificación para generar la información de forma codificada y la información de forma codificada se reconstruye como la información de forma reconstruida codificándose en orden inverso a lo largo del esquema de codificación. Mientras que los píxeles exteriores se eliminan para fijarse como píxeles transparentes, es decir, píxeles de textura no definida, los píxeles interiores restantes del bloque de campo superior o inferior reconstruido se proporcionan como píxeles de textura definida por cada bloque de campo.

En la etapa S204 se determina si cada bloque reconstruido está ubicado o no en el contorno del VOP actual, donde cada dicho bloque reconstruido contiene el bloque de campo superior reconstruido y el bloque de campo inferior reconstruido correspondiente. Dicho de otro modo, cada dicho bloque reconstruido se determina como un bloque interior, un bloque frontera o un bloque exterior, donde el bloque interior solo contiene los píxeles de textura definida, el bloque exterior solo contiene los píxeles de textura no definida y el bloque frontera contiene tanto los píxeles de textura definida como los píxeles de textura no definida. Si el bloque reconstruido se determina como un bloque interior, en la etapa S210, no se lleva a cabo el relleno y el proceso avanza hasta la etapa S211.

Si el bloque reconstruido es un bloque frontera, como se muestra en la Fig. 3A, en las etapas S221 a S224 los píxeles de textura no definida del bloque frontera se extrapolan a partir de los píxeles de textura definida del mismo para generar un bloque frontera extrapolado. En primer lugar, en la etapa S221, cada bloque frontera se divide en un bloque de campo frontera superior y en bloque de campo frontera inferior, tal y como se muestra en la Fig. 3B, presentando cada bloque de campo frontera $M/2 \times N$ píxeles de textura. En la etapa S222, los píxeles de textura no definida se rellenan por cada fila utilizando una técnica de relleno repetitivo horizontal mostrada en la Fig. 3C para generar una fila rellena para cada fila. Dicho de otro modo, los píxeles de textura no definida se rellenan repitiendo los píxeles frontera hacia las flechas, tal y como se muestra en la Fig. 3C, donde cada píxel frontera entre los píxeles de textura definida está ubicado en el contorno, es decir, en el borde, del VOP actual. Si cualquier píxel de textura no definida puede llenarse mediante la repetición de más de un píxel frontera, se utiliza el valor promedio de los valores repetidos.

Si hay una o más filas transparentes en cada bloque de campo superior o inferior, en la etapa S223, cada fila transparente se rellena utilizando una o más de las filas definidas o rellenas más próximas del bloque de campo superior o inferior correspondiente, donde la fila definida tiene en la misma todos los píxeles de textura definida. Por ejemplo, tal y como se muestra en la Fig. 3D, cada píxel de textura no definida de la fila transparente mostrada en el bloque de campo inferior se sustituye por el promedio de dos píxeles de textura definida o rellena en función de la fila rellena superior más próxima y de la fila rellena inferior más próxima, es decir, la segunda y la cuarta fila rellenas del bloque de campo inferior. Si la fila transparente está ubicada en la fila con la posición más alta o en la fila con la posición más baja, es decir, corresponde a la primera fila o la octava fila, cada dicho píxel de textura se sustituye por un píxel de textura definida o rellena de la fila rellena o definida más próxima.

Si solo existe un bloque de campo frontera transparente de los dos bloques de campo frontera, como se muestra en la Fig. 3B, en la etapa S224, el bloque de campo frontera transparente se rellena en función de otro bloque de campo frontera del bloque frontera, donde el bloque de campo frontera transparente, es decir, un bloque de campo

no definido, no tiene en el mismo ningún píxel de textura definida. Dicho de otro modo, si un bloque de campo superior es transparente, todos los píxeles de textura no definida del bloque de campo frontera transparente, es decir, el bloque de campo superior, pueden sustituirse por un valor constante P tal y como se muestra en la Fig. 3E, por ejemplo, un valor promedio de los píxeles de textura definida del bloque de campo inferior. El valor promedio de los píxeles definidos y de los rellenados del bloque de campo inferior también pueden utilizarse para llenar el bloque de campo transparente. Si fuera necesario, puede utilizarse un valor medio de 2^{L-1} de todos los valores posibles para cualquier píxel de textura en función de las características de canal, donde L es el número de bits asignados a cada píxel. Por ejemplo, si L es igual a 8, hay 256 píxeles de textura, de 0 a 255, y se determina que el valor medio es 128.

En lo que respecta a un VOP de movimiento rápido, el relleno debe extenderse además a bloques que se distribuyen fuera del VOP pero que son inmediatamente posteriores a cualquier bloque frontera. Si el bloque reconstruido se determina como el bloque exterior en la etapa S204, en la etapa S207 se determina si el bloque exterior es adyacente a cualquier bloque frontera extrapolado. Si el bloque exterior no es adyacente a ningún bloque frontera extrapolado, en la etapa 209 no se realiza ningún relleno y el proceso avanza hasta la etapa S211.

Si el bloque exterior es adyacente a uno o más bloques frontera extrapolados de manera que el bloque exterior corresponde a un bloque adyacente no definido, en la etapa S208 los píxeles de textura no definida del bloque adyacente no definido se rellenan en función de uno de los bloques frontera extrapolados para generar un bloque adyacente extrapolado para el bloque adyacente no definido, donde cada bloque frontera extrapolado tiene una parte del contorno A o B del VOP actual y cada bloque adyacente no definido se muestra como una región sombreada, tal y como se muestra en la Fig. 4. Si una pluralidad de bloques frontera extrapolados rodean el bloque adyacente no definido, se selecciona un bloque frontera extrapolado en una secuencia prioritaria hacia la izquierda, hacia arriba, hacia la derecha y hacia abajo en función del bloque adyacente no definido y, después, un borde vertical o un borde horizontal del bloque frontera extrapolado seleccionado se repite hacia la derecha, hacia abajo, hacia la izquierda o hacia arriba, donde el borde vertical o el borde horizontal es contiguo al bloque adyacente no definido. Tal y como se muestra en la Fig. 4, los bloques adyacentes no definidos JB4, JB20, JB9 Y JB2 seleccionan sus respectivos bloques frontera extrapolados hacia la izquierda, hacia arriba, hacia la derecha y hacia abajo a2, a10, a3 y a1, respectivamente; y el borde vertical derecho del bloque frontera extrapolado a2 se expande hacia la derecha para llenar el bloque adyacente no definido JB4, el borde horizontal inferior del bloque frontera extrapolado a10 se expande hacia abajo para llenar el bloque adyacente no definido JB20 y así sucesivamente. Además, bloques diagonales no definidos, tales como M1 y M2, pueden rellenarse con un valor constante, por ejemplo '128', para ser el bloque adyacente extrapolado para el bloque diagonal no definido, donde cada bloque diagonal no definido es diagonalmente adyacente al bloque frontera extrapolado y tiene todos los píxeles de textura no definida.

Tal y como se ha descrito anteriormente, en la etapa S211, los bloques de campo superior e inferior modificados, es decir, el bloque frontera extrapolado y el bloque adyacente extrapolado, así como los bloques interiores, se proporcionan como otra información de textura entrelazada anterior para el siguiente VOP actual.

REIVINDICACIONES

- 5 1. Un procedimiento para rellenar (S221-S224) un macrobloque frontera que tiene M x N píxeles de textura definida y de textura no definida, representando M un número de filas y representando N un número de columnas de los píxeles de textura, incluyendo dicho macrobloque frontera dos bloques de campo, comprendiendo el procedimiento:
- 10 (a) si el bloque de campo tiene píxeles de textura definida y de textura no definida, rellenar (S222, S223) los píxeles de textura no definida de dicho bloque de campo en función de los píxeles de textura definida de dicho bloque de campo, incluyendo dicho relleno (S222, S223) de los píxeles de textura no definida de dicho bloque de campo:
- 15 (a1) rellenar (S222) uno o más de los píxeles de textura no definida de dicho bloque de campo por cada fila para proporcionar una fila rellenada, y
- (a2) si dicho bloque de campo tiene una fila transparente que solo contiene los píxeles de textura no definida, rellenar (S223) la fila transparente en función de una o más de las filas rellenadas; y
- 20 (b) si el bloque de campo solo tiene los píxeles de textura no definida, rellenar (S224) dicho bloque de campo con un valor constante.
2. Un procedimiento para rellenar macrobloques de textura, incluyendo el macrobloque de textura el macrobloque frontera definido en la reivindicación 1 y un macrobloque no definido que solo contiene los píxeles de textura no definida, comprendiendo el procedimiento:
- 25 aplicar el procedimiento de la reivindicación 1 al macrobloque frontera para proporcionar un macrobloque frontera rellenado; y rellenar (S208) el macrobloque no definido en función del macrobloque frontera rellenado.
3. El procedimiento según la reivindicación 1 ó 2, en el que dicho relleno (S224) del bloque de campo con un valor constante incluye rellenar (S224) dicho bloque de campo con un valor constante de $2L-1$, donde L es un número de bits correspondiente al píxel de textura definida del macrobloque frontera.
- 30 4. El procedimiento según la reivindicación 1 ó 2, en el que dicho relleno (S224) del bloque de campo con un valor constante incluye rellenar (S224) dicho bloque de campo con un valor constante de 128.
- 35 5. El procedimiento según la reivindicación 1 ó 2, en el que dicho relleno (S222) de uno o más de los píxeles de textura no definida del bloque de campo por cada fila incluye rellenar (S222) los píxeles de textura no definida de una fila en función de uno o más de los píxeles de textura definida adyacentes a dichos píxeles de textura no definida de dicha fila.
- 40 6. El procedimiento según la reivindicación 1 ó 2, en el que dicho relleno (S223) de la fila transparente en función de una o más de las filas rellenadas incluye rellenar (S223) la fila transparente en función de una o más de las filas rellenadas adyacentes a dicha fila transparente.
- 45 7. Un aparato para rellenar un macrobloque frontera que tiene M x N píxeles de textura definida y de textura no definida, representando M un número de filas y representando N un número de columnas de los píxeles de textura, incluyendo dicho macrobloque frontera dos bloques de campo, comprendiendo el aparato:
- 50 (a) un circuito de relleno de bloque de campo definido para rellenar (S222, S223) los píxeles de textura no definida del bloque de campo que tiene los píxeles de textura definida y de textura no definida en función de los píxeles de textura definida de dicho bloque de campo, incluyendo el circuito de relleno de campo definido:
- 55 (a1) un circuito de relleno horizontal para rellenar (S222) uno o más de los píxeles de textura definida del bloque de campo por cada fila para proporcionar una fila rellenada, y
- (a2) un circuito de relleno de fila transparente para rellenar (S223) una fila transparente que solo contiene los píxeles de textura no definida del bloque de campo en función de una o más de las filas rellenadas; y
- 60 (b) un circuito de relleno de bloque de campo no definido para rellenar (S224) el bloque de campo que solo contiene los píxeles de textura no definida con un valor constante.
8. Un aparato para rellenar macrobloques de textura, incluyendo el macrobloque de textura el macrobloque frontera definido en la reivindicación 7 y un macrobloque no definido que solo contiene los píxeles de textura no definida, comprendiendo dicho aparato:
- 65

el aparato para rellenar un macrobloque frontera según la reivindicación 7 para proporcionar un macrobloque frontera rellenado, y
un circuito de relleno de macrobloque no definido para rellenar (S208) el macrobloque no definido en función del macrobloque frontera rellenado.

- 5
9. El aparato según la reivindicación 7 u 8, en el que el valor constante es $2L-1$, donde L es un número de bits correspondiente a los píxeles de textura definida del macrobloque frontera.
- 10
10. El aparato según la reivindicación 7 u 8, en el que el valor constante es 128.
11. El aparato según la reivindicación 8, en el que el macrobloque no definido es adyacente al macrobloque frontera.

FIG. 1

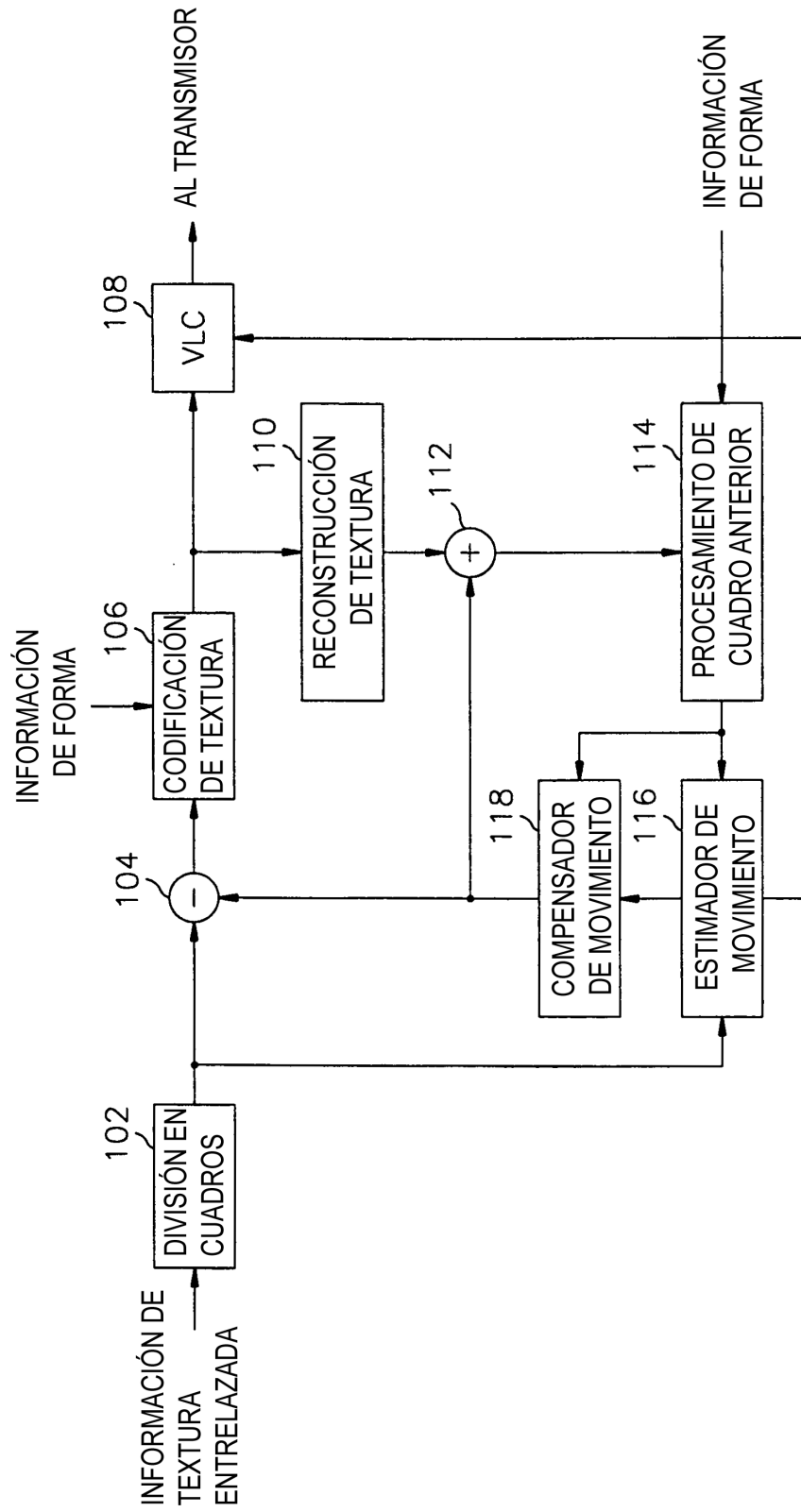


FIG. 2

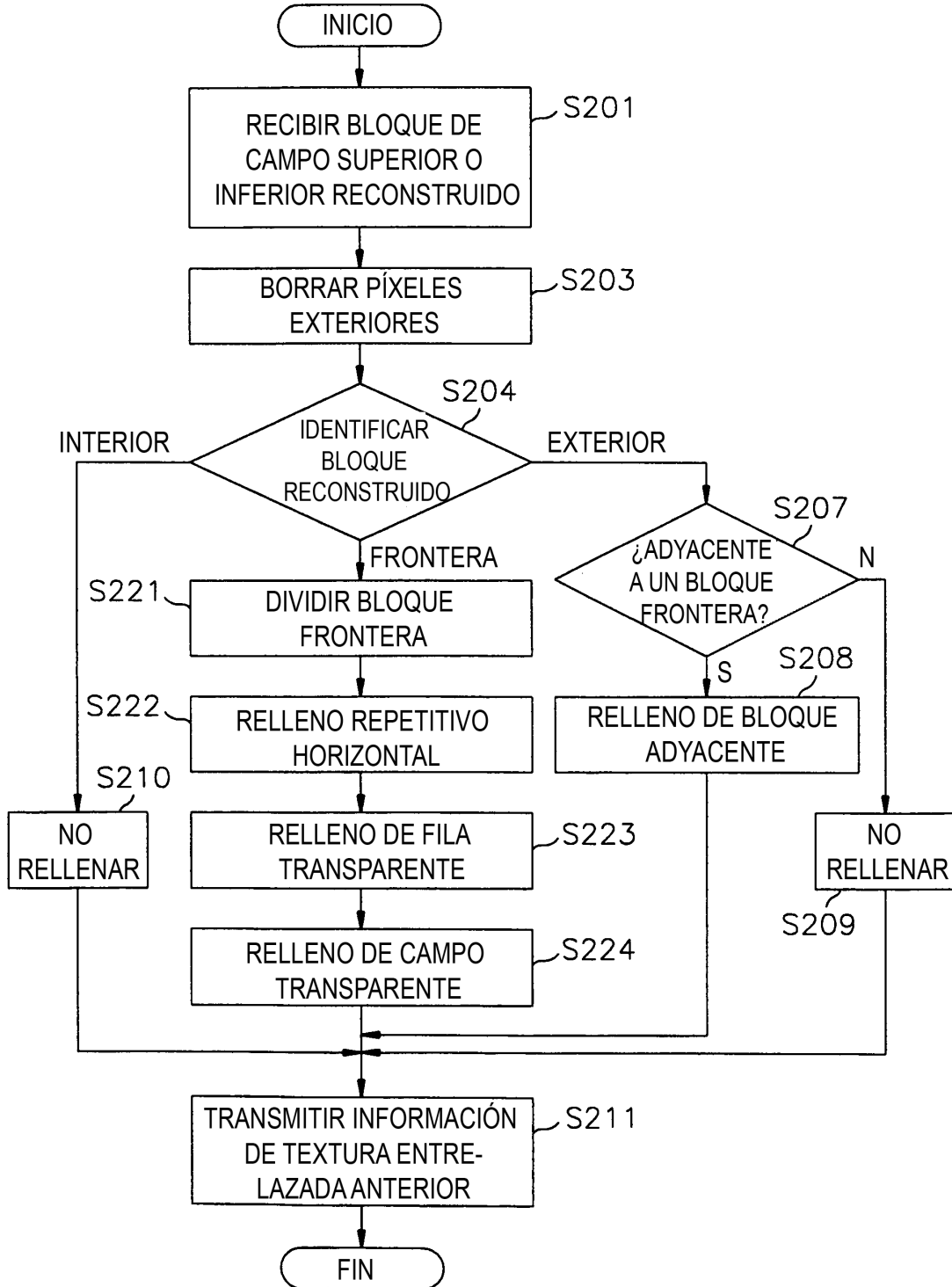


FIG. 3A

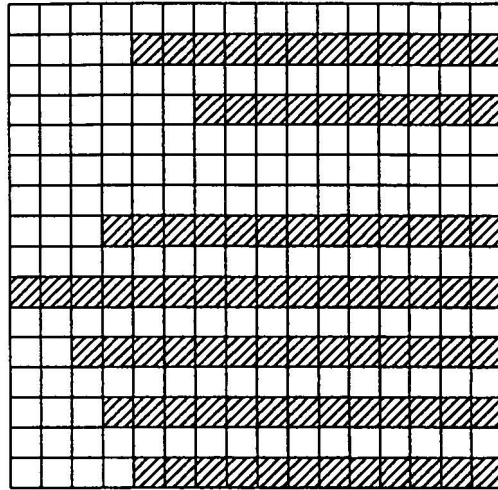


FIG. 3B

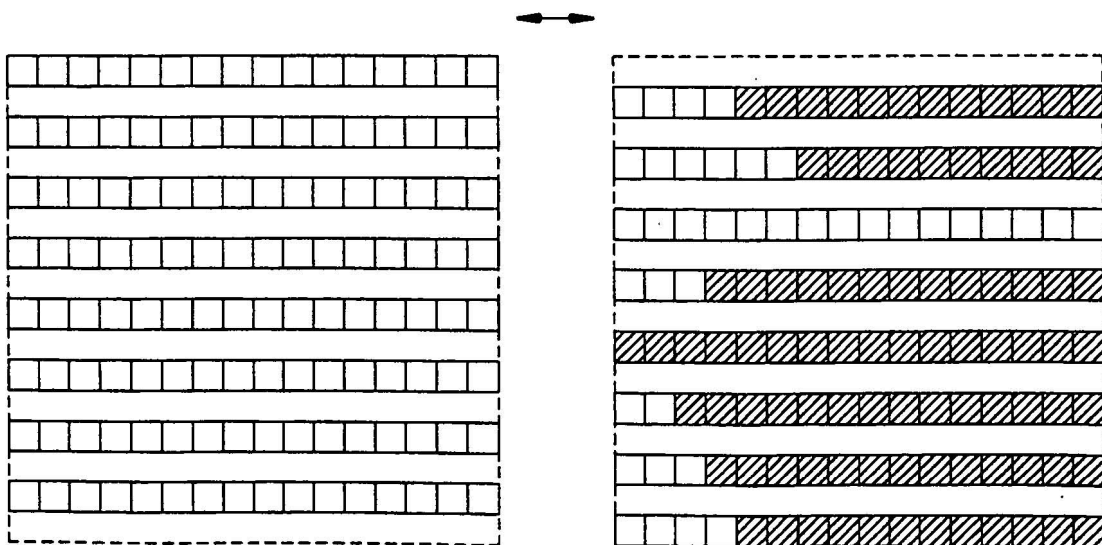


FIG. 3C

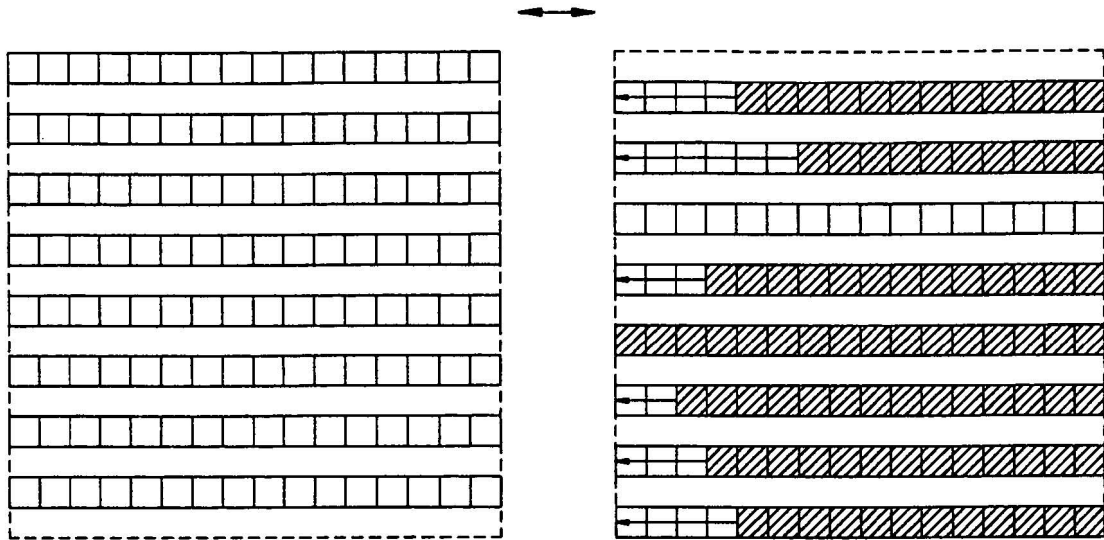


FIG. 3D

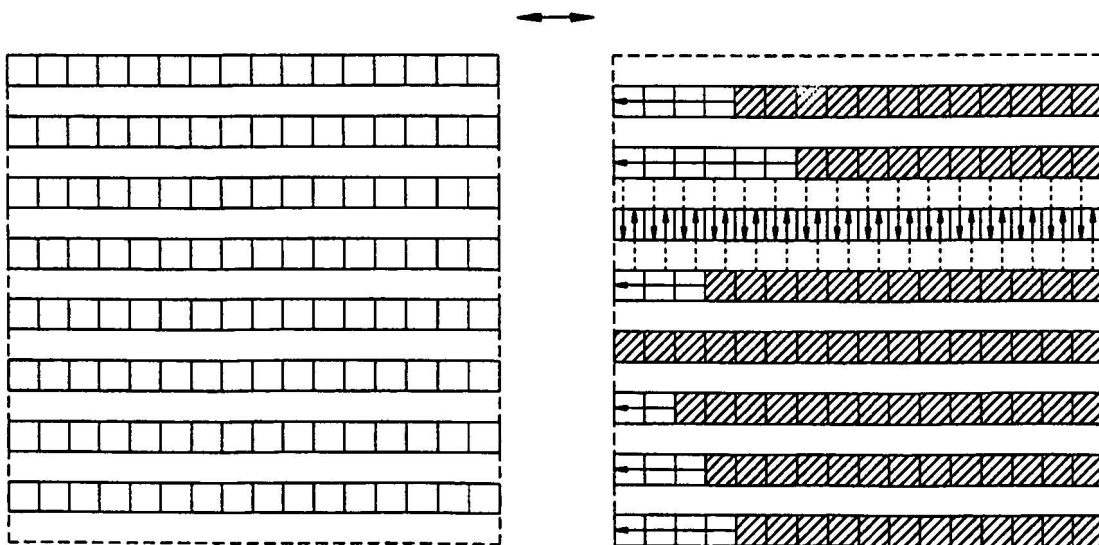


FIG. 3E

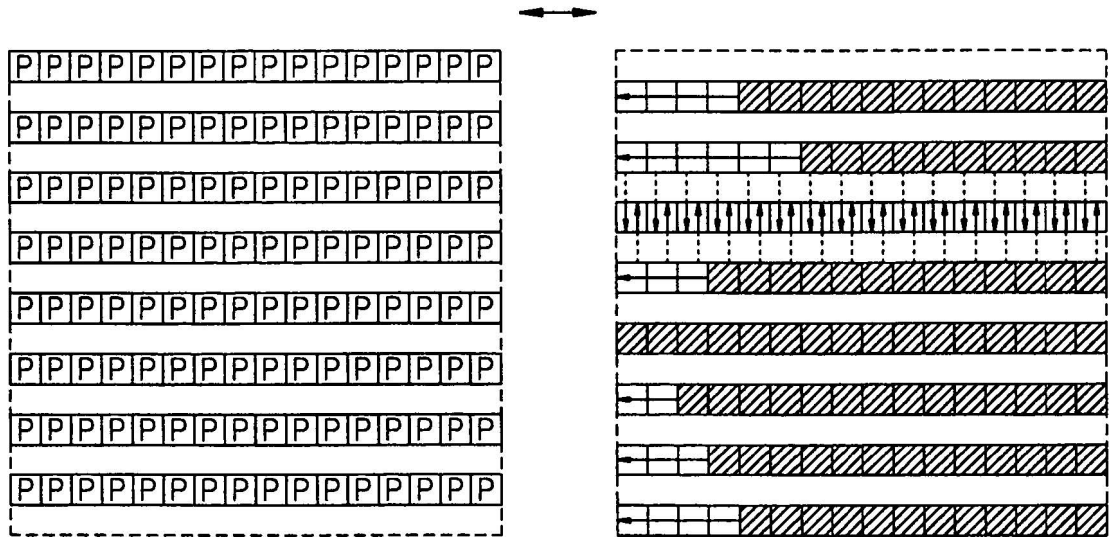


FIG. 4

