

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 457 524**

51 Int. Cl.:

H04N 7/26 (2006.01)

H04N 1/41 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **24.08.2007 E 07114947 (0)**

97 Fecha y número de publicación de la concesión europea: **15.01.2014 EP 1892964**

54 Título: **Técnica de llenado de datos para compresión de documentos compuestos**

30 Prioridad:

25.08.2006 US 509865

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

28.04.2014

73 Titular/es:

**RICOH COMPANY, LTD. (100.0%)
3-6, Nakamagome 1-chome Ohta-ku
Tokyo 143-8555, JP**

72 Inventor/es:

**FENG, GUOTONG y
GORMISH, MICHAEL J.**

74 Agente/Representante:

SUGRAÑES MOLINÉ, Pedro

ES 2 457 524 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

Técnica de llenado de datos para compresión de documentos compuestos

5 **CAMPO DE LA INVENCION**

La presente invención se refiere al campo de la compresión de documentos compuestos; más particularmente, la presente invención se refiere a una técnica de llenado de datos para su uso en la compresión de documentos compuestos.

10

ANTECEDENTES DE LA INVENCION

Contenido mixto de gráficos (MRC) es una norma ITU (T.44) para la compresión de documentos compuestos, en la que una imagen de documento se separa en múltiples capas, y cada capa se comprime mediante un codificador específico que es adecuado para esa capa. Un documento de MRC de 3 capas típico contiene una máscara binaria, una capa de primer plano y una capa de segundo plano, en el que se usa cada valor de píxel de la máscara para seleccionar entre los píxeles de primer plano y segundo plano en la reconstrucción de la imagen de gráficos por puntos. En general, un codificador de MRC de 3 capas incluye la generación de máscaras, el llenado de datos (generación de primer plano y segundo plano), ajuste a escala de primer plano y segundo plano, seguidos por una compresión separada de cada capa.

15

20

25

30

35

Aunque la generación de máscaras es crítica, el llenado de datos también es muy importante para conseguir un alto rendimiento de compresión. En general, se requiere un método de llenado de datos para un codificador basado en MRC para llenar todos los píxeles "sin importancia" (es decir, los píxeles irrelevantes para la reconstrucción) en las capas tanto de primer plano como de segundo plano. Con una compresión sin pérdidas, los píxeles "sin importancia" afectan sólo a la tasa de transmisión de bits. Sin embargo, cuando se usa una compresión con pérdidas para obtener menores tasas de transmisión de bits, una mala elección de los píxeles "sin importancia" puede cambiar los píxeles "de importancia" (es decir, los píxeles relevantes para la reconstrucción) durante la decodificación. Aunque los píxeles "sin importancia" no son directamente relevantes para la reconstrucción de la imagen, la selección de valores para estos píxeles puede tener un impacto significativo sobre el rendimiento de distorsión de tasa de la compresión. Por ejemplo, supongamos que los píxeles sin importancia en las capas de primer plano y segundo plano se llenan con un valor de color único. En este caso, pueden producirse artefactos de anillo inaceptables en la imagen descomprimida en el límite del primer plano y segundo plano con cualquier compresión con pérdidas tal como JPEG y JPEG2000, y por tanto el beneficio de la compresión de MRC se pierde significativamente.

40

45

Hay varios enfoques que se han propuesto para resolver estos problemas. Estos métodos pueden clasificarse en tres categorías: llenado de datos de dominio espacial; llenado de datos de dominio de frecuencia iterativo; y transformada dependiente de máscara.

Los enfoques de llenado de datos de dominio espacial son generalmente sencillos y rápidos. Los enfoques de llenado de datos de dominio de frecuencia iterativo proporcionan mayor compresión, pero son más caros desde el punto de vista computacional. Los enfoques de transformada dependiente de máscara dan como resultado una compresión muy alta, pero requieren una interacción entre la máscara y el primer plano/segundo plano para la compresión de las capas de primer plano y segundo plano y, por tanto, no cumplen completamente con MRC.

50

Más específicamente hay varios algoritmos de llenado de datos de dominio espacial, algunos de los cuales se describen a continuación. En un enfoque de llenado de datos de dominio espacial, se realizan múltiples pasos en cada bloque que contiene una mezcla de píxeles de importancia y sin importancia. En cada paso, cada píxel no llenado se determina por el promedio de los píxeles de importancia vecinos. Este procedimiento se repite hasta que todos los "espacios" se llenen. Para más información, véase De Querioz, "On Data-Filling Algorithms for MRC Layers", Proc. IEEE Intl. Conf. on Image Processing, ICIP, Vancouver, Canadá, vol. II, págs. 586-589, septiembre de 2000.

55

60

Otro enfoque de llenado de datos de dominio espacial realiza un método de separación de capa con correspondencia de codificador que incluye un algoritmo de llenado de datos basado en bloques de un paso para la interpolación de píxeles sin importancia. Más específicamente, para cada bloque que contiene una mezcla de píxeles de importancia y sin importancia, los valores de los píxeles sin importancia se asignan basándose en el promedio de los píxeles de importancia de ese bloque. Cada bloque sin importancia se llena por o bien un valor predefinido o bien el valor promedio del bloque llenado previo. Para más información, véase la patente estadounidense n.º 6.941.024, titulada "Coder Matched Layer Separation and Interpolation for Compression of Compound Documents", presentada en septiembre de 2005.

65

Otro enfoque de llenado de datos de dominio espacial se implementa en LuraDocuments desarrollado por Luratech. En este enfoque, se generan un primer plano y un segundo plano a una resolución reducida por un factor de 3. En cada bloque de 3x3 que contiene píxeles tanto de importancia como sin importancia, se toma un promedio de todos

los píxeles de importancia excepto los píxeles de borde. Entonces, se usa un filtro de promediado de 5 x 5 para interpolar de manera iterativa los píxeles sin importancia usando estos valores promedio hasta que se llenen todos los "espacios". Este método sí considera excluir los píxeles de borde en el promediado, reduciendo de ese modo un artefacto de anillo. Para más información, véase Thierschmann, *et al.*, "A Scalable DSP Architecture For High-Speed Color Document Compression", Document Recognition and Retrieval VIII, Kantor, *et al.*, Editors, Proceedings of SPIE vol. 4307 (2001), San José, CA, enero de 2001.

Segmentación y llenado de múltiple resolución (MSF) es un algoritmo de llenado de datos de múltiple resolución usado en el dominio de tren de ondas. El método usa una idea de "borde con pérdidas" que gestiona el efecto de borde para el llenado de datos. Para más información, véase la patente estadounidense con número de serie 10/927.323, titulada "Multi-resolution Segmentation and Fill", presentada el 25 de agosto de 2004.

Pavlidis G. *et al.*, "Compressing the background layer in compound images using JPEG and data filling" en Signal Processing: Image Communication, Elsevier Science Publishers, Ámsterdam, NL, volumen 20, n.º 5, junio de 2005, páginas 487 a 502, da a conocer un codificador de contenido adaptativo para comprimir documentos compuestos de gráficos por puntos. Puede conseguirse adaptividad de contenido empleando un enfoque por capas. En este enfoque, una imagen compuesta se segmenta en capas de modo que pueden usarse codificadores apropiados para comprimir estas capas individualmente. Se usan técnicas de llenado de datos para llenar las capas inicialmente dispersas para adaptar las características de las capas a las de los codificadores.

SUMARIO DE LA INVENCION

En la reivindicación 1 se define un método para llenado de datos según la invención.

BREVE DESCRIPCION DE LOS DIBUJOS

La presente invención se entenderá más completamente a partir de la descripción detallada proporcionada a continuación y a partir de los dibujos adjuntos de diversas realizaciones de la invención que, sin embargo, no deben limitar la invención a las realizaciones específicas, sino que sólo son para su explicación y entendimiento.

La figura 1 es un ejemplo de una imagen original y una máscara con indicadores para capas de primer plano y segundo plano.

La figura 2 es un diagrama de bloques de una realización de un sistema de compresión de documentos compuestos.

La figura 3 ilustra un ejemplo para los tres tipos de bloques de clasificación, incluyendo un bloque interior, un bloque sin importancia y un bloque de borde.

La figura 4 ilustra una ventana de 24x24 colocada en el ejemplo de la figura 3.

La figura 5 ilustra un ejemplo de un procedimiento de llenado de datos.

La figura 6 ilustra una detección de píxeles de borde en una proximidad de 3x3 en la zona de máscara.

La figura 7 ilustra un ejemplo para los cinco tipos de bloques de clasificación, incluyendo un bloque interior con píxeles de borde, un bloque interior sin píxeles de borde, un bloque sin importancia con píxeles de borde, un bloque sin importancia sin píxeles de borde y un bloque de borde.

La figura 8 ilustra los pasos secuenciales y la tabla de estado asociada de la realización de un procedimiento de llenado de datos con los tratamientos de borde especiales.

La figura 9 es un diagrama de flujo de un proceso para realizar un primer paso de una primera técnica de llenado de datos.

La figura 10 es un diagrama de flujo de un proceso para realizar un paso posterior en la primera técnica de llenado de datos.

La figura 11 es un diagrama de flujo de una realización de un proceso para realizar un primer paso en una segunda técnica de llenado de datos.

La figura 12 es un diagrama de flujo de una realización de un proceso para realizar un paso posterior en la segunda técnica de llenado de datos.

La figura 13 es un diagrama de bloques de un sistema informático a modo de ejemplo que puede realizar una o más de las operaciones descritas en el presente documento.

La figura 14 es un diagrama de bloques de una realización alternativa de un compresor de documentos compuestos.

DESCRIPCIÓN DETALLADA DE LA PRESENTE INVENCION

5 Se describe un método, un aparato y un artículo de fabricación para el llenado de datos. La técnica de llenado de datos descrita en el presente documento genera capas de imágenes para una compresión de documentos compuestos. Esta técnica crea de manera iterativa zonas alrededor de los píxeles “de importancia” en una escala basada en bloques para interpolar píxeles “sin importancia” y píxeles de borde. Esto reduce, y minimiza potencialmente, la posibilidad de discontinuidad de color cerca de las zonas de píxel de importancia, y minimiza de ese modo los artefactos de anillo debido a la compresión con pérdidas.

10 En la siguiente descripción, se exponen numerosos detalles para proporcionar una explicación más minuciosa de la presente invención. Sin embargo, resultará evidente para un experto en la técnica que la presente invención puede ponerse en práctica sin estos detalles específicos. En otros ejemplos, se muestran estructuras y dispositivos bien conocidos en forma de diagrama de bloques, en lugar de en detalle, con el fin de evitar confundir la presente invención.

15 Algunas partes de las descripciones detalladas a continuación se presentan en términos de algoritmos y representaciones simbólicas de operaciones en bits de datos dentro de una memoria informática. Estas descripciones y representaciones algorítmicas son los medios usados por los expertos en las técnicas de procesamiento de datos para transportar de la manera más eficaz lo esencial de su trabajo a otros expertos en la técnica. En este caso, y en general, se concibe que un algoritmo es una secuencia coherente en sí misma de etapas que conducen a un resultado deseado. Las etapas son aquellas que requieren manipulaciones físicas de cantidades físicas. Habitualmente, aunque no necesariamente, estas cantidades adoptan la forma de señales eléctricas o magnéticas que pueden almacenarse, transferirse, combinarse, compararse, y manipularse de otro modo. En ocasiones ha demostrado ser conveniente, principalmente por motivos de uso común, hacer referencia a estas señales como bits, valores, elementos, símbolos, caracteres, términos, números, o similares.

20 Sin embargo, debe tenerse en cuenta que todos estos términos y similares deben asociarse con las cantidades físicas apropiadas y que son etiquetas meramente convenientes aplicadas a estas cantidades. A menos que se mencione específicamente de otro modo tal como resultará evidente a partir de la siguiente discusión, se aprecia que en toda la descripción, las discusiones que utilicen términos tales como “procesar” o “computar” o “calcular” o “determinar” o “presentar visualmente” o similares, hacen referencia a la acción y procesos de un sistema informático, o dispositivo informático electrónico similar, que manipula y transforma los datos representados como cantidades físicas (electrónicas) dentro de registros y memorias del sistema informático en otros datos representados de manera similar como cantidades físicas dentro de las memorias o registros de sistema informático u otros dispositivos de almacenamiento, transmisión y presentación visual de información de este tipo.

25 La presente invención también se refiere a un aparato para realizar las operaciones en el presente documento. Este aparato puede construirse especialmente para los fines requeridos, o puede comprender un ordenador de uso general activado o reconfigurado de manera selectiva por un programa informático almacenado en el ordenador. Un programa informático de este tipo puede almacenarse en un medio de almacenamiento legible por ordenador, tal como, pero sin limitarse a, cualquier tipo de disco incluyendo discos flexibles, discos ópticos, CD-ROM y discos magnetoópticos, memorias de sólo lectura (ROM), memorias de acceso aleatorio (RAM), EPROM, EEPROM, tarjetas magnéticas u ópticas, o cualquier tipo de medio adecuado para almacenar instrucciones electrónicas, estando cada uno acoplado a un bus de sistema informático.

30 Los algoritmos y presentaciones visuales presentados en el presente documento no se refieren de manera inherente a ningún ordenador particular u otro aparato. Pueden usarse diversos sistemas de uso general con programas según las enseñanzas en el presente documento, o puede que resulte conveniente construir un aparato más especializado para realizar las etapas del método requeridas. La estructura requerida para una variedad de estos sistemas resultará evidente a partir de la descripción a continuación. Además, la presente invención no se describe con referencia a ningún lenguaje de programación particular. Se apreciará que pueden usarse una variedad de lenguajes de programación para implementar las enseñanzas de la invención tal como se describe en el presente documento.

35 Un medio legible por máquina incluye cualquier mecanismo para almacenar o transmitir información de una manera legible por una máquina (por ejemplo, un ordenador). Por ejemplo, un medio legible por máquina incluye una memoria de sólo lectura (“ROM”); memoria de acceso aleatorio (“RAM”); medios de almacenamiento de disco magnético, medios de almacenamiento óptico; dispositivos de memoria *flash*; señales eléctricas, ópticas, acústicas u otra forma de señales propagadas (por ejemplo, ondas portadoras, señales infrarrojas, señales digitales, etc.); etc.

Visión de conjunto

40 Se da a conocer una técnica de llenado de datos de múltiples pasos iterativo, basado en bloques. Esta técnica toma una máscara y una imagen original como entradas y crea unas capas de primer plano y segundo plano llenadas

como salidas. La figura 1 muestra un ejemplo de una máscara 101 y una imagen original 102, y la capa de primer plano 103 y la capa de segundo plano 104 no llenadas inicialmente. En una realización, la capa de primer plano 103 y la capa de segundo plano 104 se separan en bloques de 8x8. Sin embargo, pueden usarse otros tamaños de bloque, incluyendo tamaños de bloque que no sean cuadrados.

5 Para facilitar la siguiente descripción, ésta sólo se centra en la aplicación de las técnicas a una de las dos capas (por ejemplo, la capa de primer plano); sin embargo, debe entenderse que lo mismo se aplica a la otra capa (por ejemplo, la capa de segundo plano).

10 Para la capa de primer plano, para los fines del presente documento, todos los píxeles negros indicados en la máscara se definen como píxeles “de importancia”, o píxeles relevantes, y todos los píxeles blancos como píxeles “sin importancia”, o píxeles irrelevantes. Sólo se usarán los píxeles de importancia de la capa de primer plano para reconstruir la imagen decodificada.

15 La figura 2 es un diagrama de flujo de una realización de un compresor de documentos compuestos. Haciendo referencia a la figura 2, la imagen de gráficos por puntos 201 se introduce en la unidad de generación de máscaras 202. En respuesta a ello, la unidad de generación de máscaras 202 genera máscaras 210. Las máscaras 210 sufren una compresión sin pérdidas mediante una unidad de compresión sin pérdidas 206. En una realización, la unidad de compresión sin pérdidas/con pérdidas 206 realiza un esquema de compresión bien conocido, tal como, por ejemplo, pero sin limitarse a, JBIG, JBIG2, MMR, etc. La salida de la unidad de compresión sin pérdidas 206 son datos de máscara comprimidos.

20 La máscara 210 y la imagen de gráficos por puntos 201 también se introducen en la unidad de llenado de datos de múltiples pasos 203. En respuesta a estas entradas, la unidad de llenado de datos de múltiples pasos 203 realiza una operación de llenado de datos y genera una capa de primer plano 211 y una capa de segundo plano 212. La unidad de llenado de datos de múltiples pasos 203 realiza un proceso de llenado de datos de múltiples pasos iterativo, basado en bloques para generar una capa de segundo plano 212 y una capa de primer plano 211 como parte de la compresión de documentos compuestos.

25 En una realización, el proceso de llenado de datos de múltiples pasos iterativo, basado en bloques comprende realizar una clasificación de bloques y a continuación asignar valores a píxeles en un bloque basándose en su clasificación usando un proceso iterativo. En una realización, la clasificación de bloques de píxeles se realiza clasificando en primer lugar cada píxel en uno de dos tipos: píxel de borde y píxel no de borde, y a continuación clasificando los bloques en tipos diferentes en términos de aparición de píxeles de borde, píxeles sin importancia y píxeles de importancia en cada bloque. Un método para clasificar píxeles comprende: examinar una ventana de clasificación de píxeles centrada alrededor de cada píxel, determinar si algún píxel en la ventana tiene un valor de máscara diferente de ese píxel y clasificar ese píxel como borde si algún píxel alrededor de ese píxel en la ventana tiene un valor de máscara diferente de ese píxel. En una realización, esta ventana de clasificación de píxeles es una ventana de 3x3.

30 Después de la clasificación, se produce la asignación de valores a los bloques. La asignación de valores para cada uno de uno o más bloques (por ejemplo, bloques con bordes) de una capa (por ejemplo, una capa de primer plano o de segundo plano) se basa en un promedio de valores de píxel en una ventana que contiene ese bloque. La ventana es más grande que el tamaño de bloque. Por ejemplo, en una realización, la ventana es de 24 píxeles por 24 píxeles mientras que el tamaño de bloque es de 8 píxeles por 8 píxeles. En una realización alternativa, el tamaño de ventana es de forma rectangular con al menos un lado que no es un múltiplo del tamaño de bloque.

35 Obsérvese que la selección de tamaño de ventana se basa en uno o más de un grupo que consiste en: un algoritmo de compresión para su uso en la compresión de documentos compuestos; una transformada usada en la compresión de documentos compuestos; y contenido de la imagen.

40 Volviendo a la figura 2, en una realización, la capa de primer plano 211 se ajusta a escala mediante la unidad de ajuste a escala 204, que proporciona una capa de primer plano ajustada a escala y la capa de segundo plano 212 se ajusta a escala mediante la unidad de ajuste a escala 205.

45 Después de cualquier ajuste a escala opcional, se comprime la capa de primer plano 211 usando un compresor 207. El compresor 207 realiza una compresión con pérdidas. En una realización, el compresor 207 realiza la compresión de JPEG2000. En otra realización, el compresor 207 realiza la compresión de JPEG. El compresor 207 proporciona parte de los datos comprimidos 220.

50 Además, después de cualquier ajuste a escala opcional, se comprime la capa de segundo plano 212 usando el compresor 208. El compresor 208 realiza una compresión con pérdidas. En una realización, el compresor 208 realiza una compresión de JPEG 2000. En una realización alternativa, el compresor 208 realiza una compresión de JPEG. El compresor 208 proporciona parte de los datos comprimidos 220.

55

Ejemplo de un proceso de llenado de datos

El proceso de llenado de datos incluye la clasificación de píxeles, clasificación de bloques, procedimiento de inicialización, seguido de múltiples pasos de procesamiento de bloques de datos en una capa (por ejemplo, primer plano, segundo plano, etc.).

Con respecto a la clasificación de bloques, como inicialización, todos los bloques de 8x8 de la capa de primer plano no llenada se clasifican en uno de 3 tipos usando la imagen de máscara. Estos tres tipos incluyen un bloque interior, un bloque "sin importancia" y un bloque de borde. El bloque interior contiene sólo los píxeles de importancia, el bloque sin importancia contiene sólo píxeles sin importancia y el bloque de borde contiene una mezcla de píxeles tanto de importancia como sin importancia. La figura 3 muestra un ejemplo para los 3 tipos de bloques, en los que la letra "I", "X" y "E" indican bloques interior, sin importancia y de borde, por separado, con respecto a la imagen "R" que aparece en la capa de primer plano.

Se mantiene una tabla de estado de llenado de datos a través de todo el proceso de llenado de datos para tener un seguimiento del estado de llenado de cada bloque en cada paso. En un ejemplo, hay 3 valores definidos para el estado en la tabla:

0 - bloque no llenado

1 - bloque llenado justo en este paso

2 - bloque llenado en un paso previo

Un experto en la técnica reconocerá que pueden usarse otros esquemas de marcado.

Como parte de un procedimiento de inicialización antes del primer paso, se establece el estado de llenado de datos de cada bloque tal como se mantiene en la tabla de estado de llenado de datos para indicar que el bloque no está llenado (por ejemplo, ajustado a 0). Al final de cada paso y antes de que comience el siguiente paso, todas las ubicaciones en la tabla con un número 1 que se han llenado se cambian al número 2. Durante cada paso, el estado del bloque procesado se ajusta a 1 inmediatamente después de que se llene. La figura 5 muestra los pasos secuenciales y la tabla de estado asociada de todo el procedimiento de llenado de datos. Haciendo referencia a la figura 5, la columna izquierda muestra los pasos segmentados del procedimiento de llenado de datos en la capa de primer plano mientras que la columna derecha muestra la tabla de estado. En el paso cero, la tabla de estado indica que el estado de llenado de datos para cada bloque no está llenado. Después del primer paso, el estado de llenado de datos de la mayoría de los bloques todavía no está llenado, mientras que algunos de los bloques están llenados (mostrado con un primer tipo de sombreado). Después del segundo paso, los llenados en el paso previo se muestran con un segundo tipo de sombreado que es diferente del paso previo, mientras que se muestran algunos bloques como no llenados y se muestran otros como llenados en este paso con el primer tipo de sombreado. Finalmente, en el último paso, se muestran todos los bloques como llenados en un paso previo usando el segundo tipo de sombreado.

Después de la inicialización, se realiza un primer paso especial. En el primer paso, se procesan todos los bloques de borde y bloques internos. Para cada bloque interno, se copian los valores de píxel a partir de la imagen original y se asignan a sus correspondientes ubicaciones de píxel en el bloque y primer plano o segundo plano llenados. Para cada bloque de borde, se selecciona una ventana más grande que tiene el bloque de borde actual en su centro. En un ejemplo, la ventana es de 24x24 tal como se muestra en la figura 4. Pueden usarse otros tamaños de ventana tal como se mencionó anteriormente. Estas ventanas pueden ser de forma cuadrada (por ejemplo, de 12x12) o de forma rectangular siendo la parte superior y la parte inferior diferentes en tamaño a los dos lados. A los píxeles sin importancia del bloque actual del primer plano se les asigna entonces el valor del promedio de los píxeles de importancia en la ventana. Las entradas en el estado de llenado de datos que corresponden al bloque actual se establecen entonces para indicar que se han llenado en el paso actual (por ejemplo, 1). El valor medio del bloque actual se guarda en una memoria intermedia para su uso posterior. Al final del primer paso, todos los valores de 1 en la tabla de estado de llenado de datos se cambian a 2.

Después de realizar el primer paso especial, se realizan uno o más pasos posteriores. En los pasos posteriores, se usan los valores de los bloques de borde y bloques internos llenados para interpolar de manera iterativa los bloques sin importancia. En cada paso, para cada bloque no llenado, se comprueba en primer lugar el estado de llenado de datos de cada uno de sus 8 bloques vecinos. Si ninguno de los 8 bloques vecinos se llena en un paso previo, entonces el estado de llenado de datos del bloque actual permanece no llenado, y se salta para el paso actual. De otro modo, a cada píxel del bloque actual se le asigna el valor promedio de las medias de todos los bloques vecinos que se llenan en el paso previo. El estado de llenado de datos del bloque actual se establece en la tabla de estado de llenado de datos para indicar que el bloque actual se llenó en este paso (por ejemplo, se ajusta a 1). El valor medio del actual se guarda en la memoria intermedia de medias de bloques para el uso del siguiente paso. Al final de cada paso, todos los valores de 1 en la tabla de estado de llenado de datos se cambian a 2.

El procedimiento iterativo termina cuando se llenan todos los bloques de la capa de primer plano (y capa de segundo plano).

5 La figura 9 es un diagrama de flujo de un proceso para realizar un primer paso en un proceso de llenado de datos. Este proceso sigue la clasificación de píxeles. El proceso se realiza mediante una lógica de procesamiento que puede comprender hardware (por ejemplo, un conjunto de circuitos, lógica dedicada, etc.), software (tal como el que se ejecuta en un sistema informático de uso general o una máquina dedicada) o una combinación de ambos.

10 Haciendo referencia a la figura 9, el proceso comienza seleccionando una capa (bloque de procesamiento 901). La capa puede ser la capa de primer plano o la capa de segundo plano. A partir de la capa, la lógica de procesamiento selecciona un primer bloque (bloque de procesamiento 902). En un ejemplo, los bloques son bloques de 8x8 en las capas de primer plano o de segundo plano. Después de seleccionar el bloque, la lógica de procesamiento determina si el bloque es un bloque sin importancia (bloque de procesamiento 903). Si lo es, el proceso pasa al bloque de procesamiento 910 en el que la lógica de procesamiento evalúa si el bloque actual es el último bloque. Si lo es, entonces el paso termina. Si no, el proceso pasa al bloque de procesamiento 904 en el que la lógica de procesamiento selecciona el siguiente bloque y el proceso continúa en el bloque de procesamiento 903. En el bloque de procesamiento 903, si el bloque no es un bloque sin importancia, la lógica de procesamiento evalúa si el bloque es un bloque interior (bloque de procesamiento 905). Si lo es, la lógica de procesamiento llena el bloque con los valores de píxel originales a partir de la imagen original (bloque de procesamiento 906), establece el estado de llenado de bloque del bloque para indicar que se llenó en este paso (bloque de procesamiento 907) y el proceso pasa al bloque de procesamiento 910 en el que continúa el proceso.

25 Si el bloque no es un bloque interior, el procesamiento pasa al bloque de procesamiento 908, en el que la lógica de procesamiento llena los píxeles de importancia en el bloque con los valores de píxel originales a partir de la imagen original y llena los píxeles sin importancia en el bloque con un promedio de todos los píxeles de importancia en una zona de ventana más grande que el bloque. En un ejemplo, la ventana es una ventana de 24x24, aunque pueden seleccionarse otras ventanas. A continuación, la lógica de procesamiento establece el estado de llenado de bloque para indicar que el bloque se llenó en este paso (bloque de procesamiento 909) y el proceso pasa al bloque de procesamiento 910 en el que continúa el proceso.

30 La figura 10 es un diagrama de flujo de un proceso para realizar un paso posterior en el ejemplo del proceso de llenado de datos. El proceso se realiza mediante lógica de procesamiento que puede comprender hardware (por ejemplo, un conjunto de circuitos, lógica dedicada, etc.), software (tal como el que se ejecuta en un sistema informático de uso general o una máquina dedicada) o una combinación de ambos.

35 Haciendo referencia a la figura 10, el proceso comienza seleccionando una capa (bloque de procesamiento 1001). En un ejemplo, la lógica de procesamiento selecciona la capa de primer plano o la capa de segundo plano. A partir de la capa, la lógica de procesamiento selecciona un primer bloque (bloque de procesamiento 1002). En un ejemplo, el bloque es un bloque de 8x8. Pueden usarse otros tamaños de bloque.

40 La lógica de procesamiento evalúa entonces si el bloque es un bloque no llenado (bloque de procesamiento 1003). Si no lo es, el proceso pasa al bloque de procesamiento 1008 en el que la lógica de procesamiento evalúa si el bloque actual es el último bloque. Si lo es, el proceso termina. Si no, el proceso pasa al bloque de procesamiento 1004 en el que la lógica de procesamiento selecciona el siguiente bloque y el proceso continúa desde el bloque de procesamiento 1003. En el bloque de procesamiento 1003, si el bloque es un bloque no llenado, la lógica de procesamiento comprueba si hay alguno de los bloques vecinos llenados en cualquier paso previo (bloque de procesamiento 1005). Si no, el proceso pasa al bloque de procesamiento 1008 en el que continúa el proceso. Si uno o más de los bloques vecinos se han llenado en un paso previo, la lógica de procesamiento llena el bloque con el promedio de las medias de los valores de todos los bloques vecinos que se han llenado en pasos previos (bloque de procesamiento 1006) y establece el estado de llenado de bloque a indicado que el bloque se llenó en el paso actual (bloque de procesamiento 1007), y luego el proceso pasa al bloque de procesamiento 1008 en el que continúa el proceso.

55 Realización de llenado de datos con tratamientos de borde especiales

Una mejora de un proceso de llenado de datos previo considera el efecto de borde en el promediado para la interpolación de bloque. Las distinciones entre el ejemplo previo y esta realización se describen a continuación.

60 Con respecto a la clasificación de bloques, como operación inicial, se detectan todos los píxeles de borde en la imagen de máscara binaria para su uso en tratamientos especiales en la interpolación de bloque. Tal como se muestra en la figura 6, se detecta un píxel como borde si tiene el valor de máscara diferente (es decir, negro o blanco) de cualquiera de sus vecinos en una ventana de tamaño predeterminado (por ejemplo, una ventana de 3x3). En una realización, el proceso de clasificación de bloques clasifica los bloques en cinco categorías de tipo de bloque que incluyen un bloque interior con píxeles de borde, un bloque interior sin píxeles de borde, un bloque sin importancia con píxeles de borde, un bloque sin importancia sin píxeles de borde y un bloque de borde (con píxeles

sin importancia, de borde y de importancia). Las definiciones del bloque interior, bloque sin importancia y bloque de borde son las mismas que las descritas anteriormente.

5 La figura 7 ilustra un ejemplo para los 5 tipos de bloques, en los que la letra "U", "I", "V", "X" y "E" corresponden a un bloque interior con píxeles de borde, un bloque interior sin píxeles de borde, un bloque sin importancia con píxeles de borde, un bloque sin importancia sin píxeles de borde y un bloque de borde, respectivamente. Obsérvese que el bloque A contiene todos los píxeles de importancia y el bloque B contiene todos los píxeles sin importancia. Sin embargo, ambos contienen píxeles de borde a lo largo del límite de bloque.

10 Tal como se describió anteriormente, se usa una tabla de estado de llenado de datos para realizar un seguimiento del estado de llenado de los bloques en la capa. El procedimiento de definición y actualización en esta realización son los mismos que en el ejemplo descrito anteriormente. La figura 8 muestra los pasos secuenciales y la tabla de estado asociada de todo el procedimiento de llenado de datos con los tratamientos de borde especiales descritos a continuación. Tal como puede observarse, la figura 8 muestra que la zona alrededor de los bordes es mucho más suave que en la figura 5, lo que indica que los artefactos de anillo pueden reducirse de manera significativa aplicando los tratamientos de borde especiales. Haciendo referencia a la figura 8, la columna izquierda muestra los pasos segmentados del procedimiento de llenado de datos en la capa de primer plano mientras que la columna derecha muestra la tabla de estado. En el paso cero, la tabla de estado indica que el estado de llenado de datos para cada bloque es no llenado. Después del primer paso, el estado de llenado de datos de la mayoría de los bloques sigue siendo no llenado, mientras se muestra que más de los bloques deben llenarse (mostrado con un primer tipo de sombreado). Después del segundo paso, los llenados en el paso previo se muestran con un segundo tipo de sombreado que es diferente del paso previo, mientras que algunos bloques se muestran no llenados y otros se muestran llenados en este paso con el primer tipo de sombreado. Finalmente, en el último paso, todos los bloques se muestran llenados en un paso previo usando el segundo tipo de sombreado.

25 Con respecto al primer paso especial, se considera el efecto de borde en la interpolación de bloque. En este paso, se procesan los bloques con tipo "I", "E", "U" y "V". Específicamente, cuando se asignan valores basándose en los valores de píxel en la ventana (por ejemplo, la ventana de 24x24) tal como se muestra en la figura 4, a cada píxel de borde o píxel sin importancia en el bloque actual se le asigna el valor promedio de todos los píxeles de importancia no de borde en la ventana. Este valor promedio se guarda en la memoria intermedia de medias de bloques. En el caso en el que hay sólo píxeles de borde en el bloque y no hay píxeles de importancia no de borde en la ventana, los valores asignados a todos los píxeles de importancia en el bloque actual son los valores en las correspondientes ubicaciones de píxel en la imagen original. En otras palabras, no se realiza un promediado. El estado de llenado de datos también se establece para indicar que el bloque no está llenado (por ejemplo, 0) para el bloque actual.

35 En los pasos posteriores, sólo se considera un tratamiento especial. Para cada bloque no llenado, todos los píxeles de importancia se mantienen sin cambios si cualquiera, y todos los otros píxeles (píxeles sin importancia) en el bloque se interpolan de la misma manera que en el ejemplo descrito anteriormente. El promedio de los píxeles interpolados también se guarda en la memoria intermedia de medias de bloques.

40 De manera similar, este proceso de llenado de datos termina de la misma manera que el ejemplo descrito anteriormente.

45 La figura 11 es un diagrama de flujo de la realización de un proceso para realizar el primer paso del proceso de llenado de datos. El proceso se realiza mediante lógica de procesamiento que puede comprender hardware (por ejemplo, un conjunto de circuitos, lógica dedicada, etc.), software (tal como el que se ejecuta en un sistema informático de uso general o una máquina dedicada) o una combinación de ambos.

50 Haciendo referencia a la figura 11, el proceso comienza mediante la lógica de procesamiento seleccionando una capa (bloque de procesamiento 1101). La lógica de procesamiento puede seleccionar o bien la capa de primer plano o bien la capa de segundo plano. La lógica de procesamiento selecciona entonces un primer bloque de la capa seleccionada (bloque de procesamiento 1102). En una realización, la lógica de procesamiento selecciona un bloque de 8x8. Sin embargo, pueden usarse otros tamaños de bloque.

55 Después de seleccionar un bloque, la lógica de procesamiento evalúa si el bloque actual es un bloque sin importancia sin píxeles de borde (bloque de procesamiento 1103). Si lo es, el proceso pasa al bloque de procesamiento 1114 en el que la lógica de procesamiento evalúa si el bloque actual es el último bloque. Si lo es, el proceso termina. Si no, la lógica de procesamiento selecciona el siguiente bloque (bloque de procesamiento 1105) y pasa al bloque de procesamiento 1103 en el que continúa el proceso. Si el bloque no es un bloque sin importancia sin píxeles de borde, la lógica de procesamiento evalúa si el bloque actual es un bloque interior sin píxeles de borde (bloque de procesamiento 1104). Si lo es, la lógica de procesamiento llena el bloque actual con los valores a partir de las correspondientes ubicaciones de píxel en la imagen original (bloque de procesamiento 1106) y establece el estado de llenado de bloque para indicar que el bloque se llenó en este paso (bloque de procesamiento 1107) y el proceso pasa al bloque de procesamiento 1114 en el que continúa el proceso.

65

Si el bloque actual no es un bloque interior sin píxeles de borde, la lógica de procesamiento pasa al bloque de procesamiento 1108 en el que la lógica de procesamiento cuenta el número de píxeles de importancia no de borde en una ventana más grande. En una realización, la ventana es de 24 píxeles x 24 píxeles. Sin embargo, puede usarse otros tamaños de ventana que sean más grandes que el tamaño de bloque. Después de contar el número de píxeles de importancia no de borde en la ventana más grande, la lógica de procesamiento evalúa si hay algún píxel de importancia no de borde en el bloque actual (bloque de procesamiento 1109). Si no lo hay, la lógica de procesamiento llena sólo los píxeles de borde en el bloque actual con los valores de las correspondientes ubicaciones de píxel en la imagen original (bloque de procesamiento 1110) y establece el estado de llenado de bloque para indicar que el bloque actual no está llenado en este paso (bloque de procesamiento 1111), y el proceso pasa al bloque de procesamiento 1114 en el que continúa el proceso. Si hay uno o más píxeles de importancia no de borde en el bloque actual, la lógica de procesamiento llena los píxeles de importancia no de borde con los valores de las correspondientes ubicaciones de píxel en la imagen original, llena todos los demás píxeles en el bloque con un promedio de todos los píxeles de importancia no de borde en la ventana de 24x24 (bloque de procesamiento 1112) y establece el estado de llenado de bloque para indicar que el bloque actual se llenó en el paso (bloque de procesamiento 1113), y el proceso pasa al bloque de procesamiento 1114 en el que continúa el proceso.

La figura 12 es un diagrama de flujo de una realización de un proceso para realizar un paso posterior en la realización que corresponde a la figura 11 del proceso de llenado de datos. El proceso se realiza mediante la lógica de procesamiento que puede comprender hardware (por ejemplo, un conjunto de circuitos, lógica dedicada, etc.), software (tal como el que se ejecuta en un sistema informático de uso general o una máquina dedicada) o una combinación de ambos.

Haciendo referencia a la figura 12, el proceso comienza seleccionando una capa (bloque de procesamiento 1201). En una realización, la lógica de procesamiento puede seleccionar la capa de primer plano o la capa de segundo plano. A partir de la capa, la lógica de procesamiento selecciona un primer bloque (bloque de procesamiento 1202). En una realización, el bloque es un bloque de 8x8. Pueden usarse otros tamaños de bloque.

Entonces la lógica de procesamiento evalúa si el bloque actual es un bloque no llenado (bloque de procesamiento 1203). Si no lo es, el proceso pasa al bloque de procesamiento 1208 en el que la lógica de procesamiento evalúa si el bloque actual es el último bloque. Si lo es, el paso termina. Si no, el proceso pasa al bloque de procesamiento 1204 en el que la lógica de procesamiento selecciona el siguiente bloque y el proceso continúa desde el bloque de procesamiento 1203. Si el bloque es un bloque no llenado, la lógica de procesamiento comprueba si alguno de los bloques vecinos se ha llenado en un paso previo (bloque de procesamiento 1205). Si no, el proceso pasa al bloque de procesamiento 1208 en el que continúa el proceso. Si hay uno o más bloques vecinos que se han llenado durante un paso previo, la lógica de procesamiento llena los píxeles sin importancia en el bloque actual con el promedio de las medias de todos los píxeles de los bloques vecinos llenados en el paso previo (bloque de procesamiento 1206) y establece el estado de llenado de bloque a "llenado en este paso" (bloque de procesamiento 1207), y el proceso pasa al bloque de procesamiento 1208 en el que continúa el proceso.

Realizaciones alternativas

Variación de paso de número final

Tal como se describió anteriormente, el proceso de llenado de datos puede tardar varios pasos hasta finalizar. Por ejemplo, si sólo hay un bloque de borde en la fila superior de la imagen (y no hay bloques interiores), entonces es necesario un paso para cada fila de bloques en la imagen con el fin de determinar el contenido del bloque en la parte inferior que queda de la página. Aunque estos pasos serán de baja complejidad computacional, para algunas aplicaciones es importante no pasar a través de la memoria múltiples veces. Por tanto, en una realización, después de cierto número fijo de pasos, puede realizarse un paso "final". En el paso final, cada bloque que tiene píxeles no determinados se ajusta a la media de bloque de valores de píxel del bloque más cercano con píxeles determinados. Si hay más de un único bloque más cercano, al bloque se le asigna al valor promedio de las medias de bloque de todos los bloques más cercanos. En una realización, se usa el valor de punto medio de nivel de gris en el paso final de modo que no es necesario que se determine el bloque más cercano con píxeles determinados. En otra realización, al bloque se le asigna un valor predeterminado.

Aunque el uso del bloque de píxel determinado más cercano en el paso final introduce cierta asimetría en la imagen, esto se produce lejos de cualquier píxel de borde, y por tanto no afectará a los valores reconstruidos.

Combinación de llenado de datos y ajuste a escala de imagen

Tal como se muestra en la figura 2, las capas de primer plano y segundo plano se ajustan a escala opcionalmente (normalmente se submuestran) tras el proceso de llenado de datos. En este caso, las operaciones de ajuste a escala de imagen y llenado de datos son independientes una de otra. Mientras que el método de llenado de datos actual puede operar de esta manera, puede obtenerse un rendimiento mejorado cuando se combina la operación de llenado de datos con la operación de ajuste a escala.

La figura 14 es un diagrama de bloques de un procesador de este tipo. Para combinar la operación de llenado de datos con el ajuste a escala, se asigna una memoria intermedia 222 para la imagen llenada al tamaño ajustado a escala en lugar del tamaño de imagen original. Puesto que el ajuste a escala es habitualmente un ajuste a escala descendente, será una memoria intermedia más pequeña. Cuando se copian los "píxeles interiores" (es decir, píxeles de importancia no de borde) a una memoria intermedia de llenado 222 se ajustan a escala basándose en el factor de ajuste a escala 1401. Esta operación de ajuste a escala sólo opera en un bloque de datos cada vez. Cuando se determinan píxeles sin importancia, no es necesario un ajuste a escala, todo el bloque ajustado a escala se llena simplemente con el valor promedio.

Cuando se realiza un ajuste a escala, en una realización, se cambia el tamaño de bloque. Por ejemplo, si se usa un ajuste a escala descendente por un factor de 3 (un factor muy común), es útil usar un tamaño de bloque de 9 por 9. De esta manera, cada bloque corresponde a un bloque de 3x3 de píxeles ajustados a escala.

La combinación de un ajuste a escala y llenado de datos tiene varias ventajas. Por ejemplo, el ajuste a escala y llenado de datos combinado reduce la complejidad computacional, y no es necesario que el algoritmo de llenado de datos determine más píxeles de los que se usarán por el sistema de compresión, no es necesario un ajuste a escala para píxeles que se llenan (al menos un 50% de los píxeles se llenará con algún valor promedio cuando se consideran tanto el primer plano como el segundo plano). El tamaño de bloque puede elegirse basándose en el factor de ajuste a escala. Además puede mejorarse la calidad de primer plano y segundo plano porque el sistema puede optimizarse con respecto a la calidad incluyendo un ajuste a escala, en lugar de optimizarlos independientemente.

La descripción en el presente documento se refiere al uso de un primer plano, segundo plano y una máscara, pero en algunos sistemas de compresión se usan múltiples capas. En el caso de múltiples capas, la reconstrucción podría proceder tal como sigue: en primer lugar la imagen actual se ajusta a la imagen de segundo plano. Entonces, para cada capa adicional, hay un objeto de máscara y un objeto de imagen. Los píxeles del objeto de imagen se combinan para dar la imagen actual basándose en el objeto de máscara. Esto se describe en detalle en la norma JPM (ISO/IEC 15444-6).

Para esta descripción, es importante darse cuenta de que la máscara que corresponde con cada objeto de imagen es la que se usa para determinar los "píxeles de importancia" y "píxeles sin importancia" y "píxeles de borde". Para la imagen de segundo plano, no hay ninguna máscara almacenada en el archivo comprimido, de hecho, se determinan los píxeles sin importancia combinando las máscaras de todas las demás capas (cualquier píxel que sea un píxel de importancia en una capa, no es un píxel de importancia en el segundo plano).

También es posible usar una máscara de escala de gris en algunos esquemas de compresión de documentos por capas incluyendo JPM. En este caso, las dos capas se combinan en proporción al valor de la máscara. Con el fin de determinar píxeles de importancia y sin importancia para determinar cómo llenar píxeles, es necesario crear una máscara binaria. Una máscara binaria puede crearse a partir de una máscara de escala de gris de varias maneras conocidas para los expertos en la técnica. El método más sencillo es la aplicación de valor umbral a la máscara, es decir, convertir todos los valores por encima de algún valor a 1 y todos los demás a cero. Obsérvese que puede ser deseable usar máscaras diferentes para el primer plano y el segundo plano. Por ejemplo, puede ser deseable que la máscara de escala de gris tenga un valor medio de por ejemplo 128, para que el píxel sea un píxel "de importancia" tanto en el primer plano como en el segundo plano, esto conducirá a una alta calidad. Alternativamente, un píxel de máscara con un valor medio de 128 podría considerarse "sin importancia" tanto en el primer plano como en el segundo plano, esto conducirá a una alta compresión.

Un ejemplo de un sistema informático

La figura 13 es un diagrama de bloques de un sistema informático a modo de ejemplo que puede realizar una o más de las operaciones descritas en el presente documento. Haciendo referencia a la figura 13, el sistema informático 1300 puede comprender un sistema informático de cliente o servidor a modo de ejemplo. El sistema informático 1300 comprende un mecanismo de comunicación o bus 1311 para comunicar información, y un procesador 1312 acoplado con el bus 1311 para procesar información. El procesador 1312 incluye un microprocesador, pero no se limita a un microprocesador, tal como, por ejemplo, Pentium™, PowerPC™, Alpha™, etc.

El sistema 1300 comprende además una memoria de acceso aleatorio (RAM) u otro dispositivo de almacenamiento dinámico 1304 (denominado memoria principal) acoplado al bus 1311 para almacenar información e instrucciones que van a ejecutarse por el procesador 1312. La memoria principal 1304 también puede usarse para almacenar variables temporales u otra información intermedia durante la ejecución de instrucciones por el procesador 1312.

El sistema informático 1300 también comprende una memoria de sólo lectura (ROM) y/u otro dispositivo de almacenamiento estático 1306 acoplado al bus 1311 para almacenar información estática e instrucciones para el procesador 1312, y un dispositivo de almacenamiento de datos 1307, tal como un disco magnético o disco óptico y su correspondiente unidad de disco. El dispositivo de almacenamiento de datos 1307 se acopla al bus 1311 para

almacenar información e instrucciones.

5 El sistema informático 1300 puede acoplarse además a un dispositivo de visualización 1321, tal como un tubo de rayos catódicos (CRT) o una pantalla de cristal líquido (LCD), acoplado al bus 1311 para presentar visualmente información a un usuario informático. Un dispositivo de entrada alfanumérico 1322, que incluye teclas alfanuméricas y otras teclas, también pueden acoplarse al bus 1311 para comunicar información y selecciones de comando al procesador 1312. Un dispositivo de entrada de usuario adicional es un control de cursor 1323, tal como un ratón, una bola de control de cursor, panel táctil, estilete o teclas de dirección de cursor, acoplado al bus 1311 para comunicar información de dirección y selecciones de comando al procesador 1312, y para controlar el movimiento de cursor en la pantalla 1321.

10 Otro dispositivo que puede acoplarse al bus 1311 es un dispositivo de copia impresa 1324, que puede usarse para marcar información en un medio tal como papel, película o tipos similares de medios. Otro dispositivo que puede acoplarse al bus 1311 es una capacidad de comunicación por cable/inalámbrica 1325 para la comunicación con un teléfono o dispositivo manual.

15 Obsérvese que cualquiera o todos los componentes del sistema 1300 y hardware asociado pueden usarse en la presente invención. Sin embargo, puede apreciarse que otras configuraciones del sistema informático pueden incluir algunos o todos los dispositivos.

20 Aunque muchas alteraciones y modificaciones de la presente invención resultarán sin duda evidentes para un experto habitual en la técnica después de leer la descripción anterior, debe entenderse que no se pretende que ninguna realización particular mostrada y descrita a modo de ilustración se considere limitativa. Por tanto, no se pretende que las referencias a detalles de diversas realizaciones limiten el alcance de las reivindicaciones que en sí mismas indican sólo aquellas características consideradas como esenciales para la invención.

25

REIVINDICACIONES

1. Método que comprende:
- 5 recibir una imagen y una máscara asociada con la imagen para generar un documento compuesto; y
- realizar un proceso de llenado de datos de múltiples pasos iterativo, basado en bloques para llenar una capa como parte de una compresión de documentos compuestos, comprendiendo la capa píxeles de importancia relevantes para la reconstrucción y píxeles sin importancia irrelevantes para la reconstrucción,
- 10 clasificar bloques de al menos una de las capas de primer plano y segundo plano,
- en el que la clasificación de bloques comprende identificar cada bloque que tiene sólo píxeles de importancia con píxeles de borde, cada bloque que tiene sólo píxeles de importancia sin píxeles de borde, cada bloque que contiene sólo píxeles sin importancia con píxeles de borde, cada bloque que contiene sólo píxeles sin importancia sin píxeles de borde y cada bloque que contiene un borde creado por al menos un píxel de importancia que es adyacente a al menos un píxel sin importancia, en el que un píxel se detecta como píxel de borde si tiene un valor de máscara diferente de cualquiera de sus vecinos;
- 15 y la realización de un proceso de llenado de datos de múltiples pasos iterativo basado en bloques comprende, para cada bloque durante un primer paso:
- llenar todos los píxeles de importancia no de borde con sus valores originales, asignar cada píxel de borde y píxel sin importancia en cada bloque clasificado como que contiene sólo píxeles de importancia con píxeles de borde, que contiene sólo píxeles de importancia sin píxeles de borde, que contiene sólo píxeles sin importancia con píxeles de borde o que contiene un borde creado por al menos un píxel de importancia que es adyacente a al menos un píxel sin importancia con un valor basándose en un promedio de píxeles de importancia no de borde en una ventana que contiene cada uno de dichos bloques, en el que la ventana es más grande que el tamaño de bloque; y
- 20 asignar cada píxel de importancia en cada bloque que contiene sólo píxeles sin importancia sin píxeles de borde a los correspondientes valores de píxel del correspondiente bloque en la imagen.
2. Método según la reivindicación 1, en el que la realización de un proceso de llenado de datos de múltiples pasos iterativo basado en bloques comprende además realizar un paso posterior para cada bloque no llenado que incluye:
- 35 mantener cada uno de dichos bloques no llenados sin llenar si ninguno de un conjunto de bloques vecinos se llena en un paso inmediatamente anterior; y
- 40 asignar cada píxel de cada uno de dichos bloques no llenados con un valor que consiste en un promedio de las medias de bloques vecinos que se habían llenado en el paso inmediatamente anterior.
3. Método según la reivindicación 2, que comprende además repetir la realización del paso posterior para cada bloque no llenado hasta que todos los bloques de la capa se hayan llenado.
- 45 4. Método según la reivindicación 2, que comprende además:
- repetir la realización del paso posterior para un número establecido de pasos; y
- 50 después de realizar el número establecido de pasos, asignar cada píxel no determinado en un bloque en la capa a un valor en un grupo que consiste en un promedio de valores de píxel en otro bloque más cercano al bloque y un valor de punto medio de nivel de gris.
- 55 5. Método según la reivindicación 1, en el que la realización del proceso de llenado de datos de múltiples pasos iterativo basado en bloques comprende mantener una tabla de estado de llenado de datos para tener un seguimiento del estado de llenado de cada bloque en cada paso del proceso de llenado de datos de múltiples pasos.
- 60 6. Método según la reivindicación 1, en el que la máscara es una máscara binaria que se genera a partir de una máscara de nivel de gris.
7. Método según la reivindicación 1, en el que la realización del proceso de llenado de datos de múltiples pasos iterativo, basado en bloques comprende adaptar un tamaño de bloque basándose en uno o más de un grupo que consiste en qué paso está realizándose y el contenido de la imagen que corresponde al
- 65

bloque.

- 5 8. Método según la reivindicación 1, que comprende además realizar un ajuste a escala combinado con el proceso de llenado de datos de múltiples pasos iterativo, basado en bloques mediante el que se usa una memoria intermedia para una versión de tamaño a escala de una imagen llenada que tiene un tamaño menor que el tamaño de la imagen.

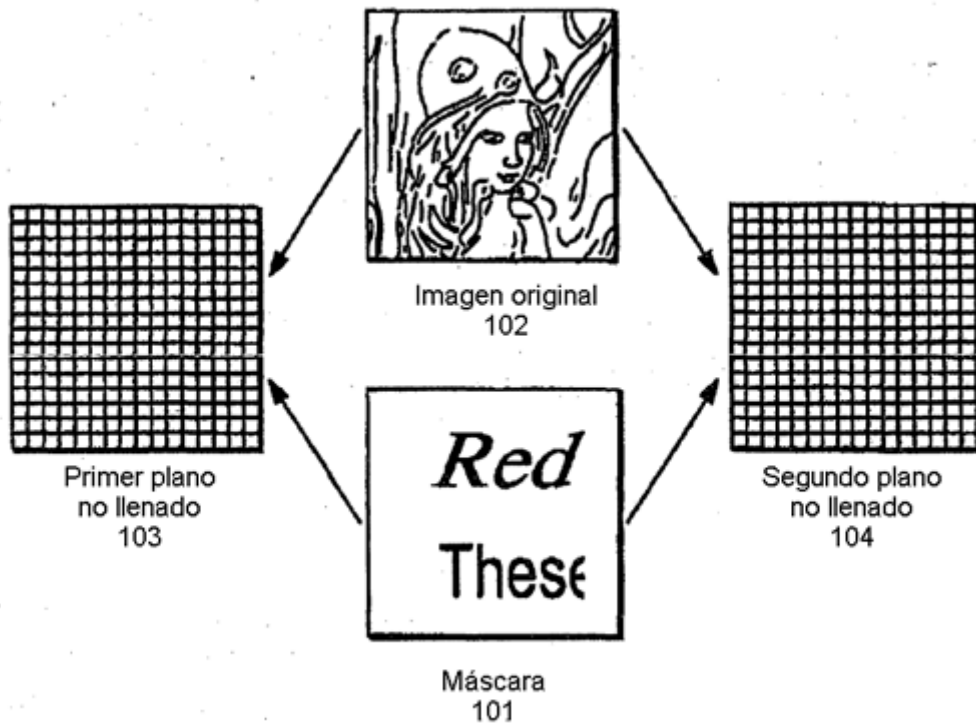


FIG. 1

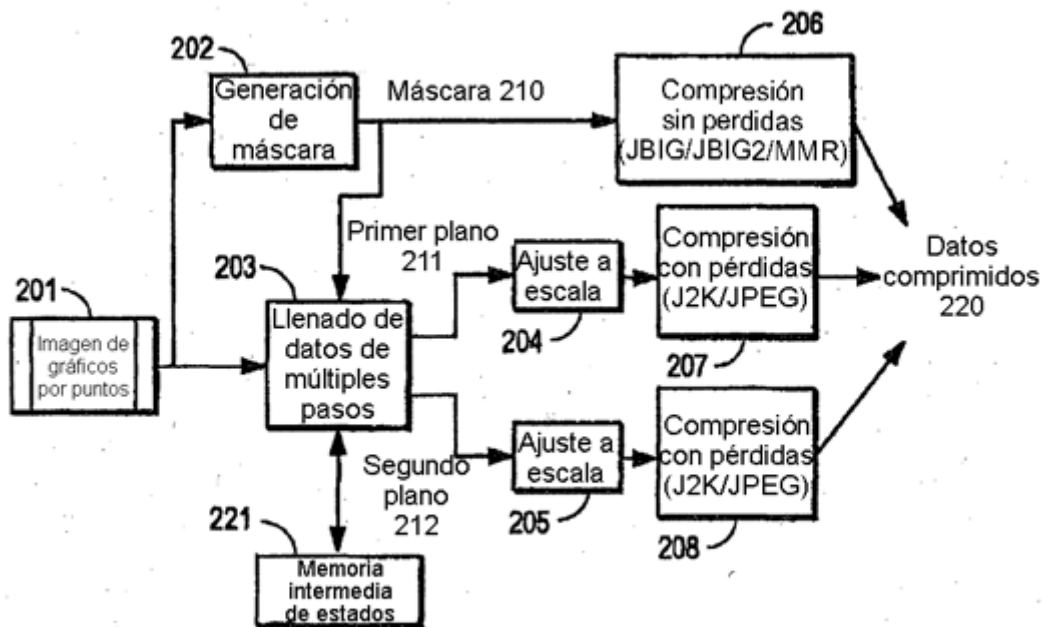


FIG. 2

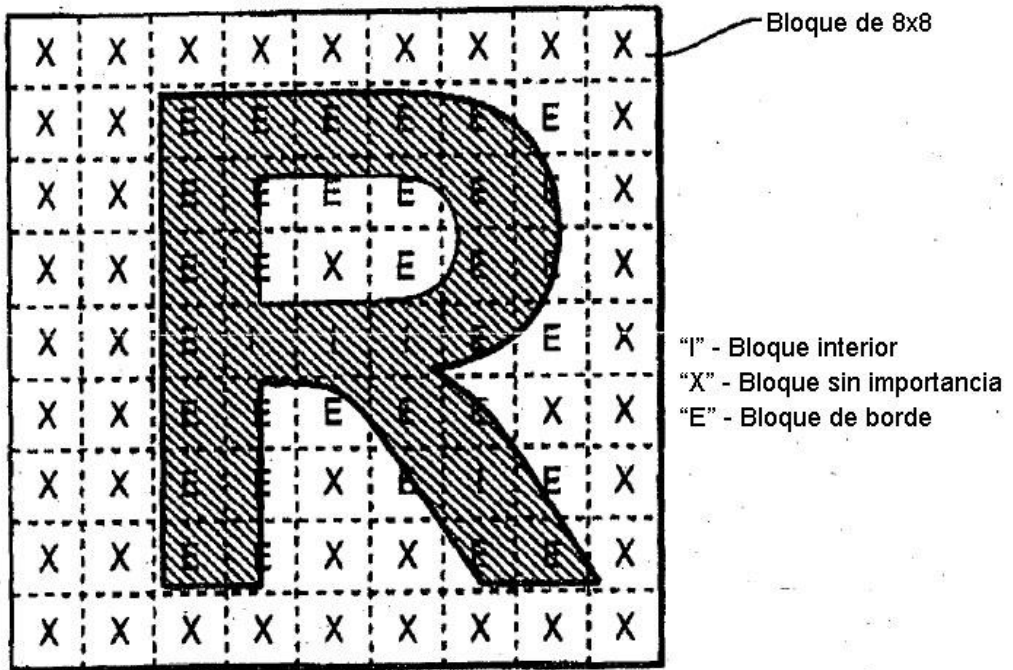


FIG. 3

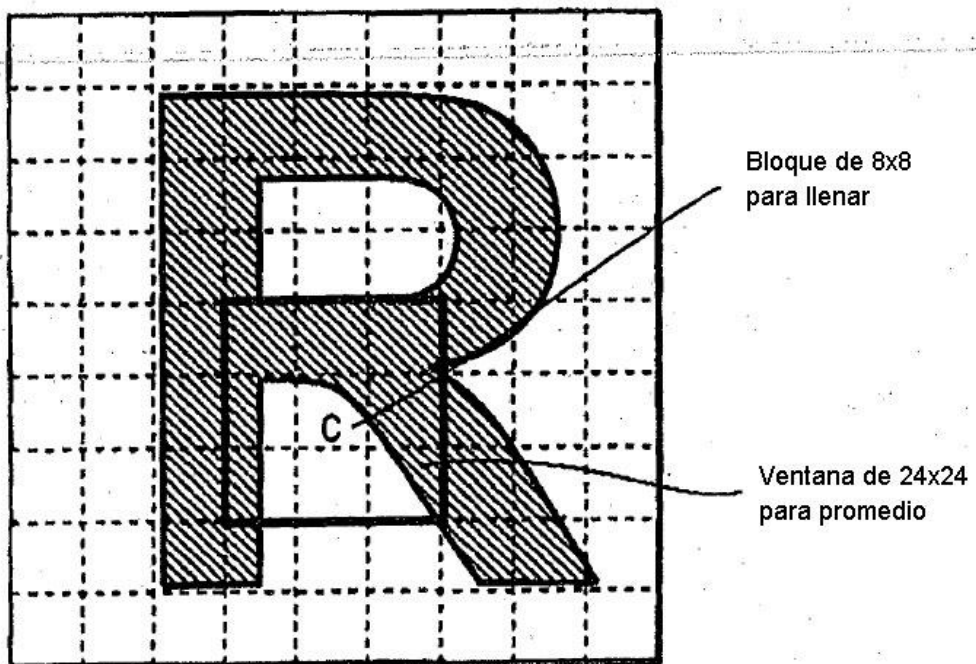


FIG. 4

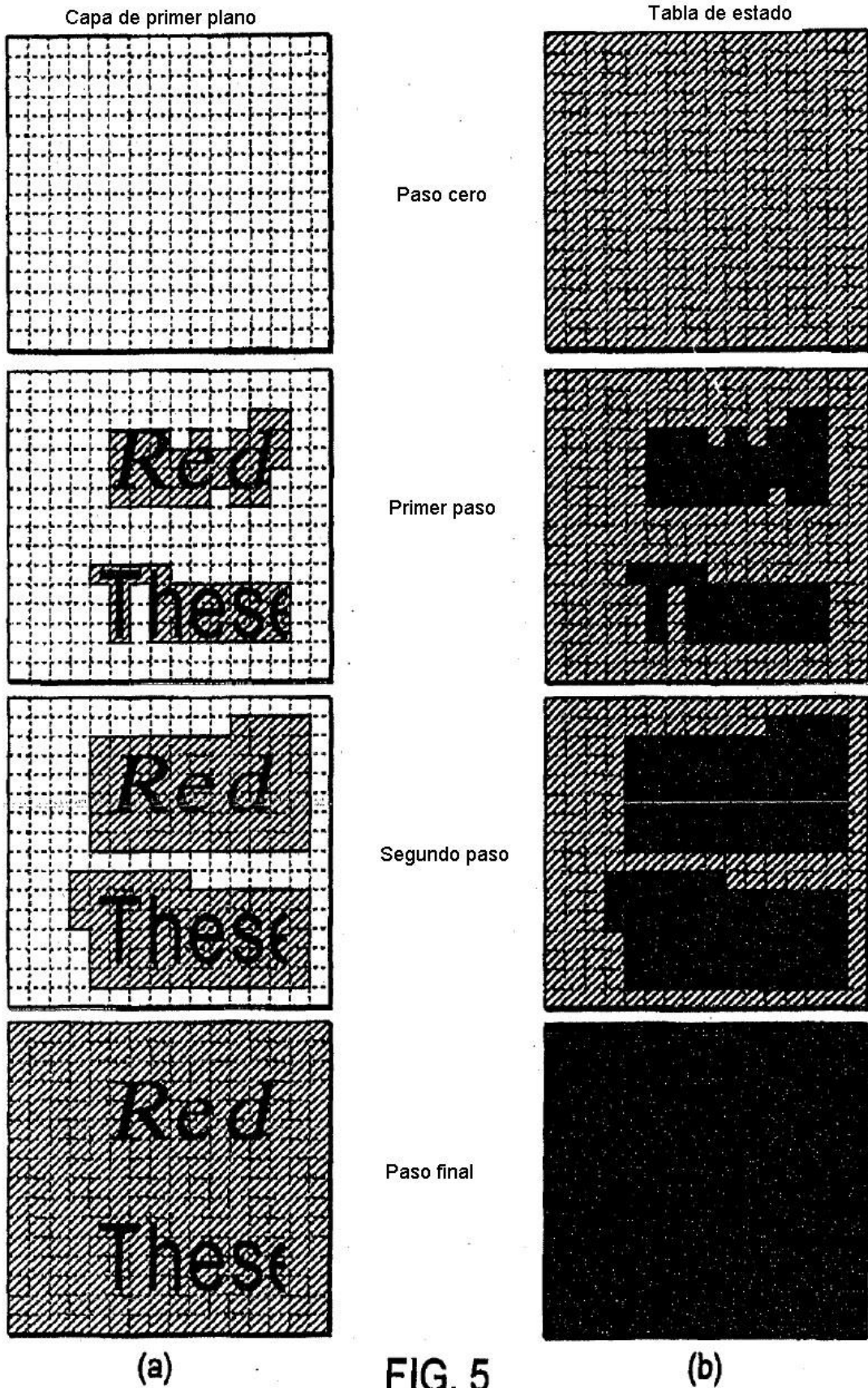


FIG. 5

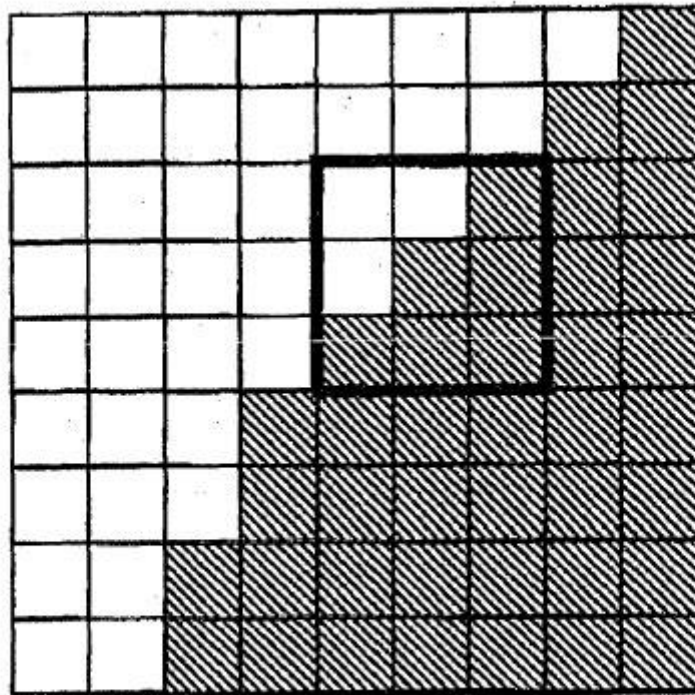


FIG. 6

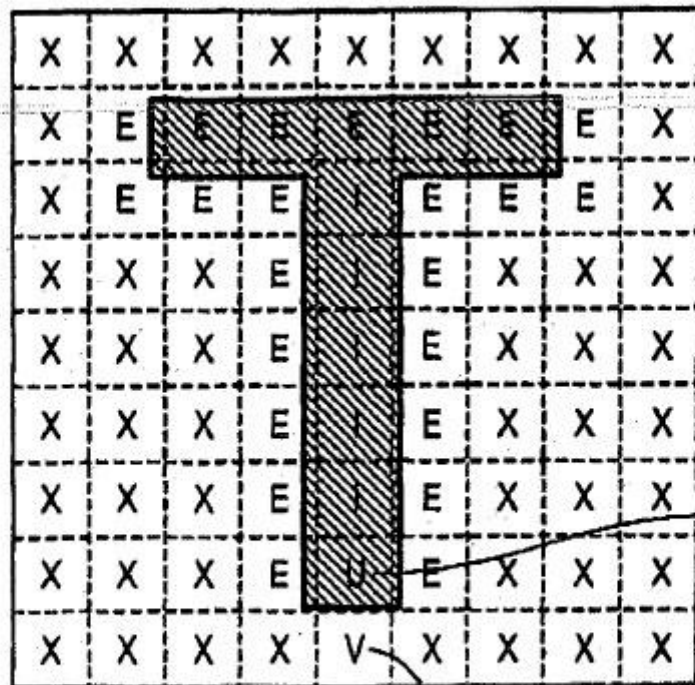
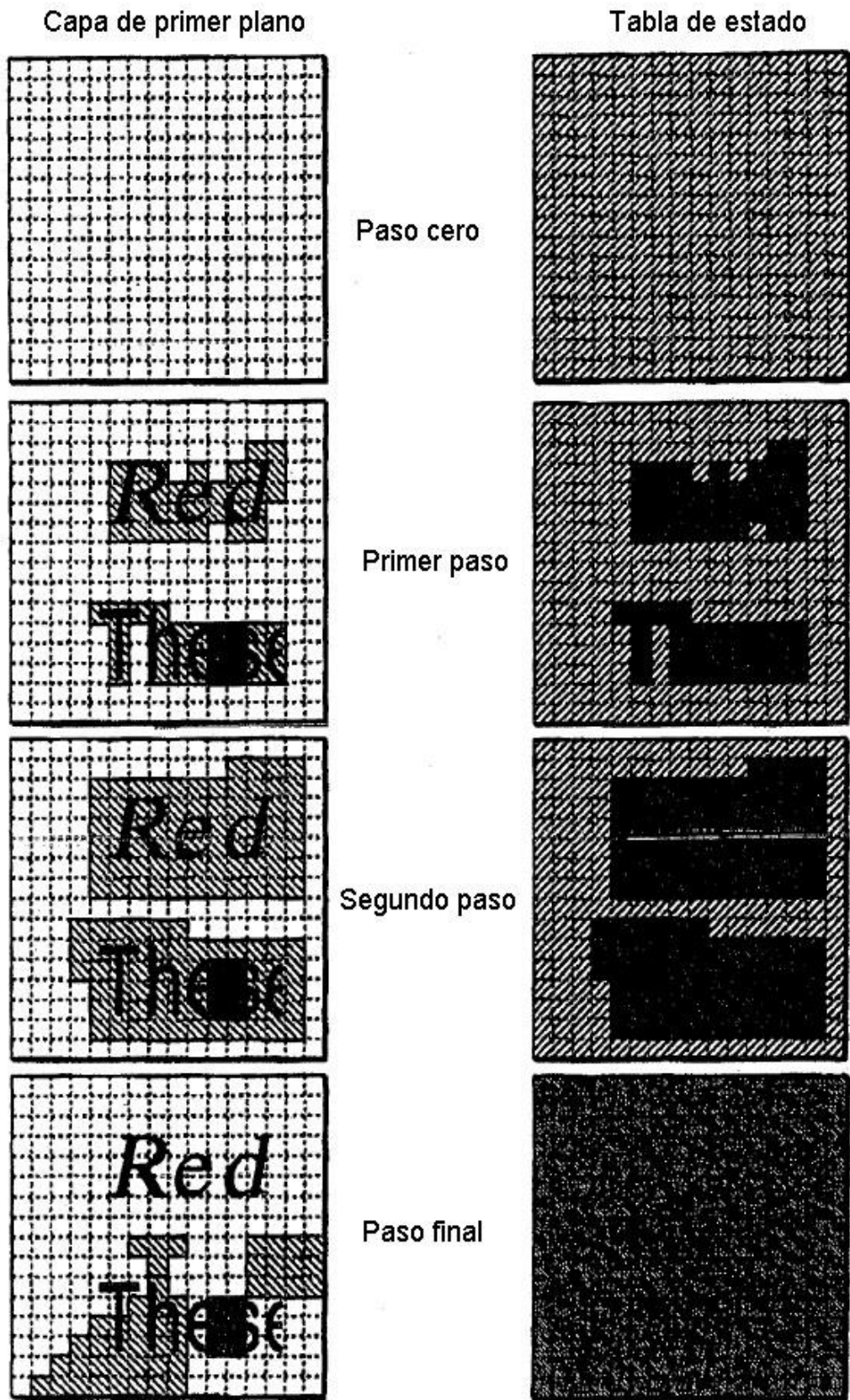


FIG. 7



(a)

FIG. 8

(b)

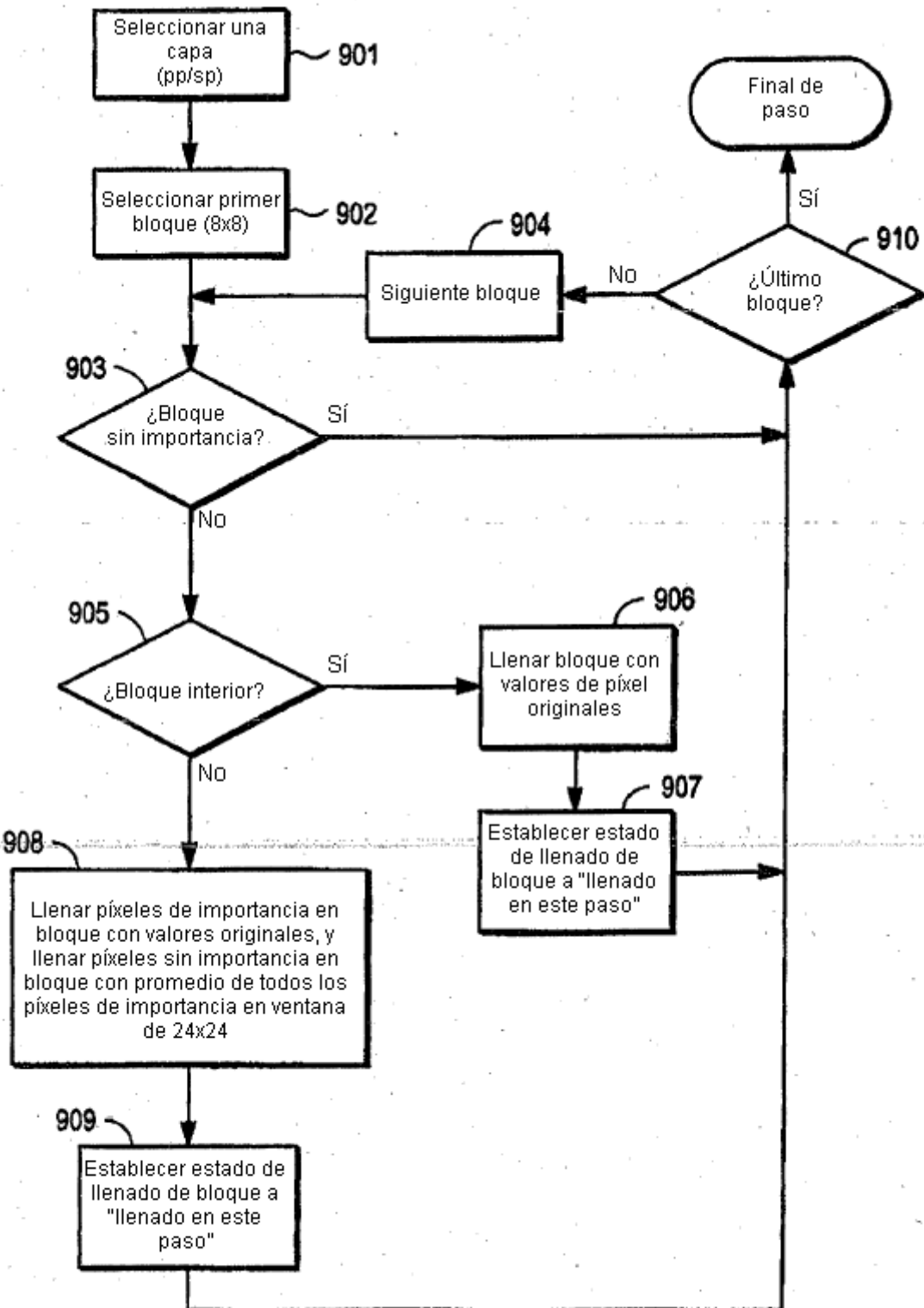


FIG. 9

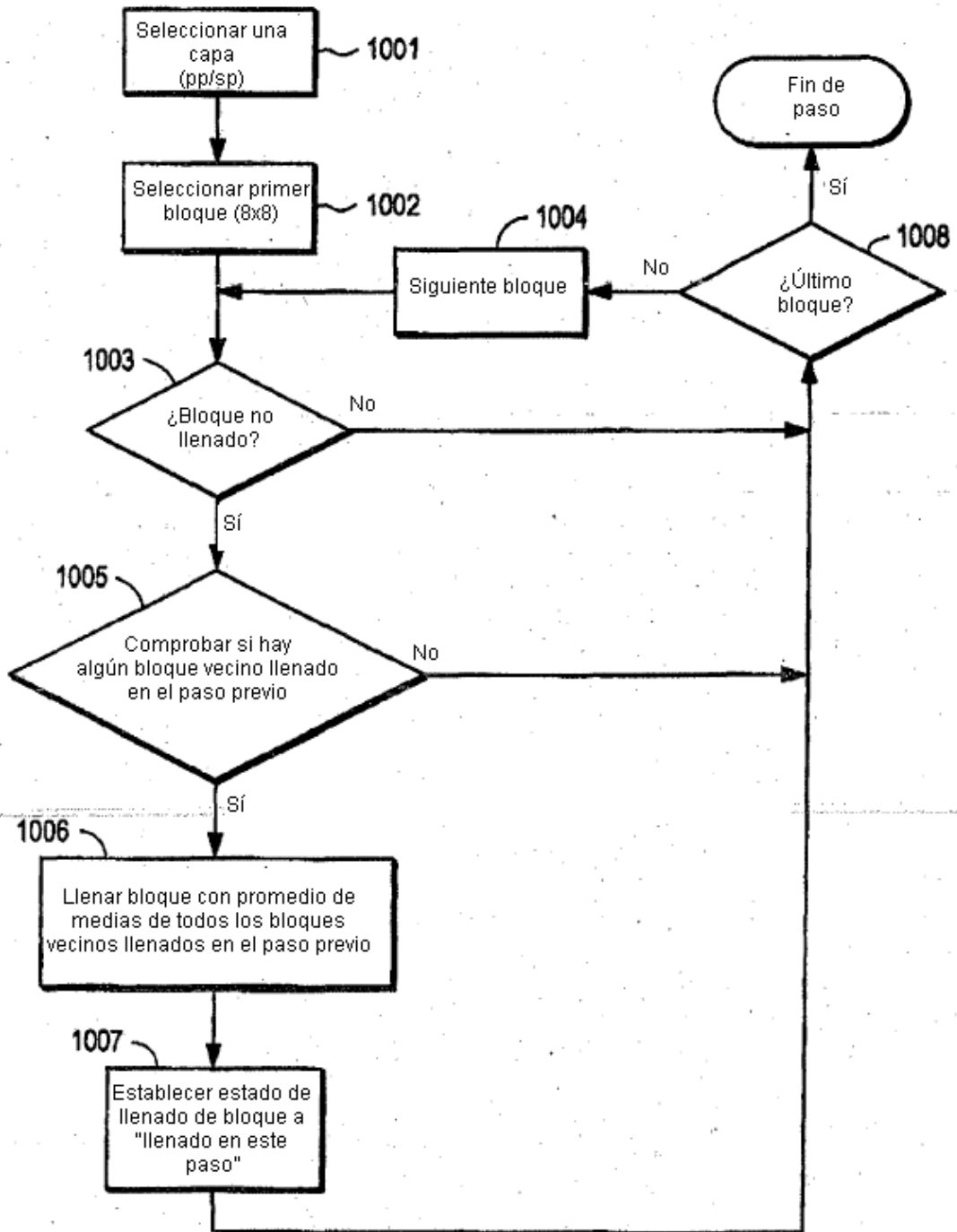


FIG. 10

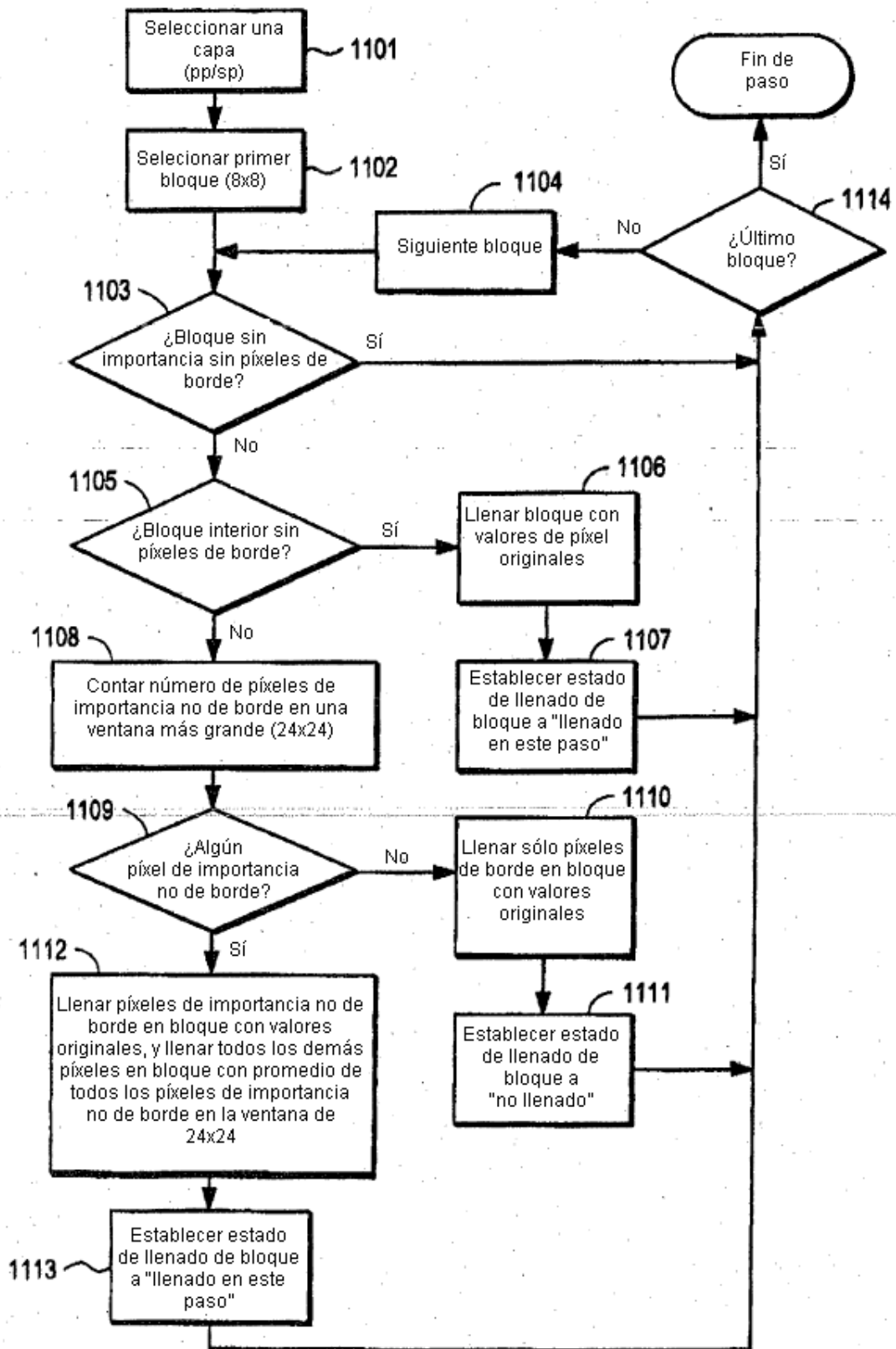


FIG. 11

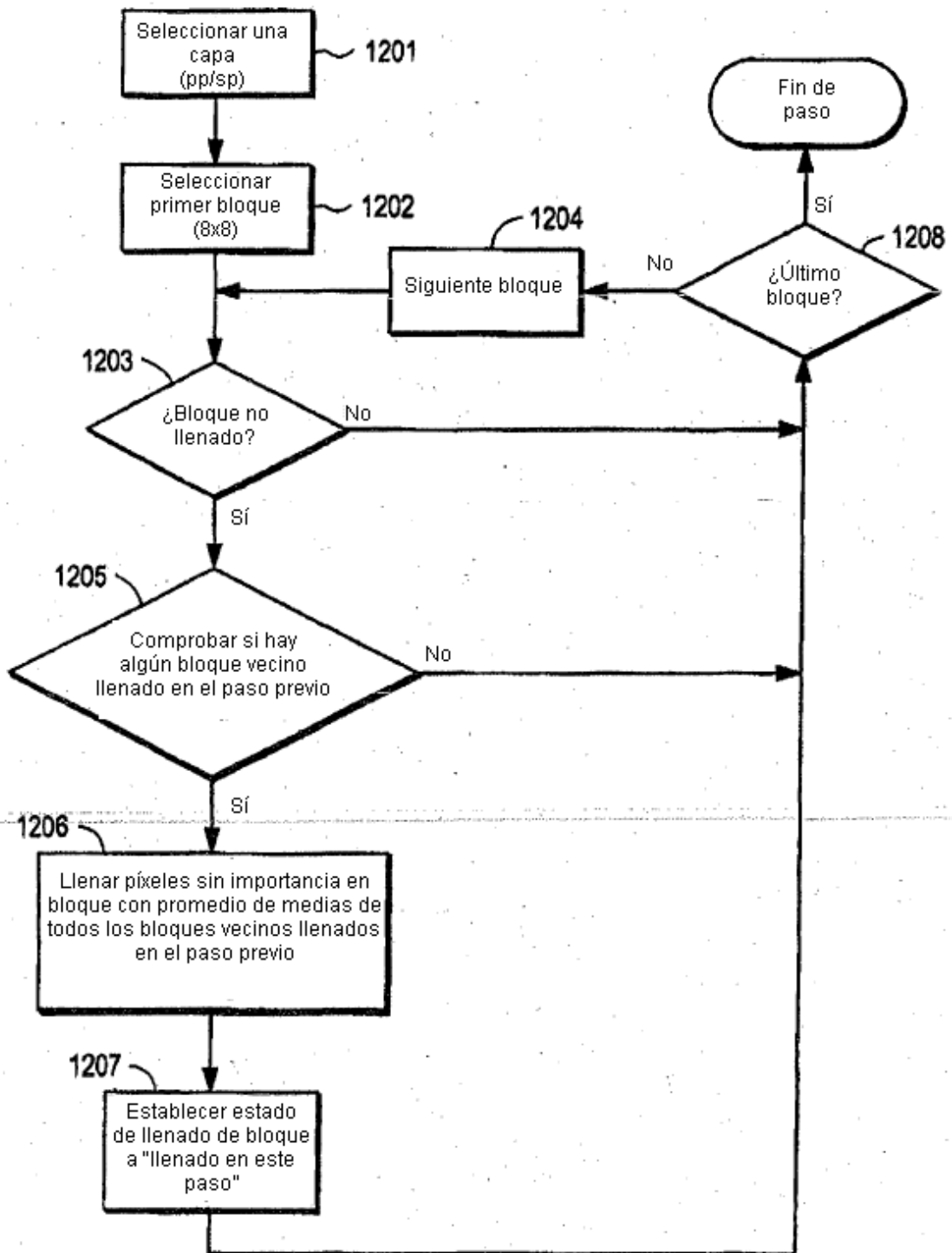


FIG. 12

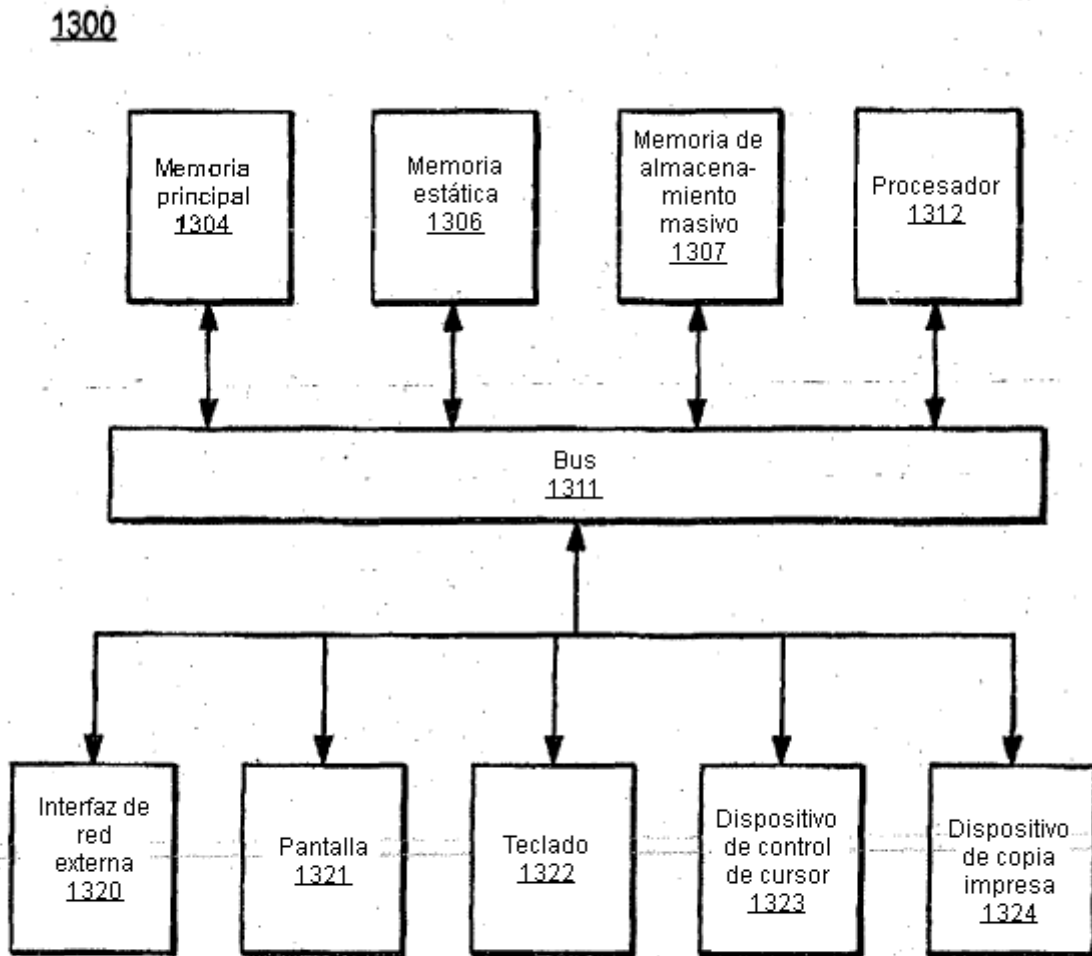


FIG. 13

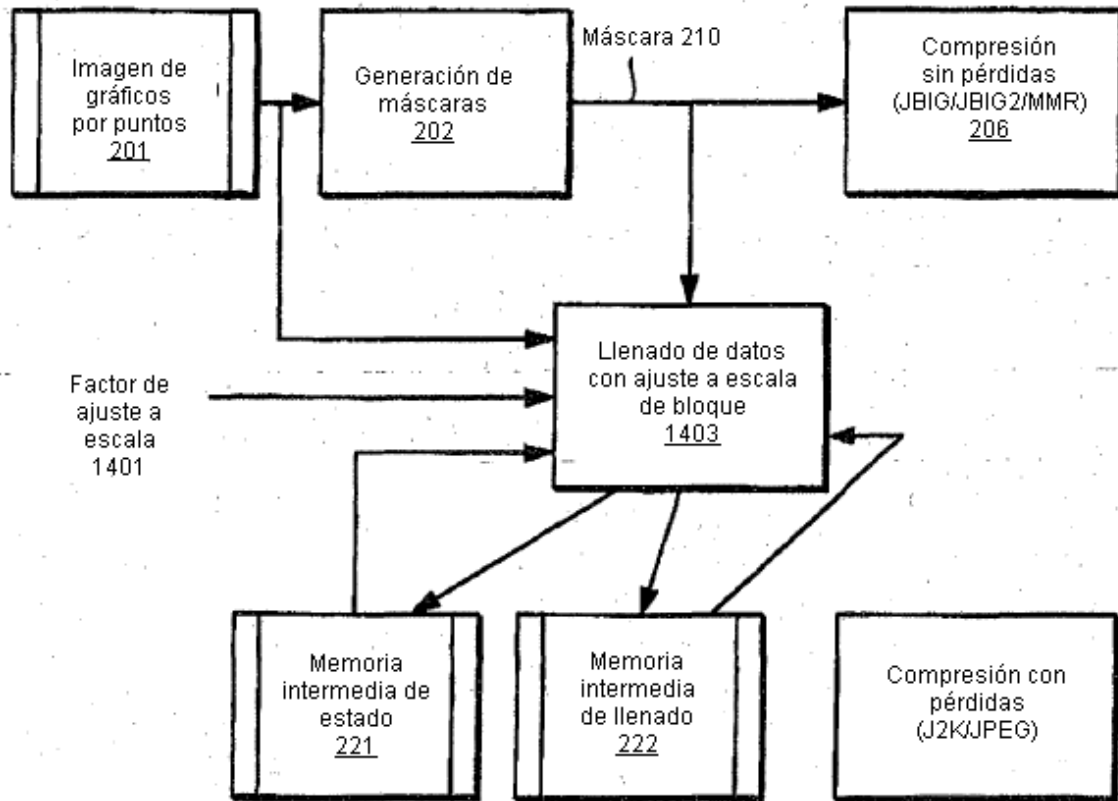


FIG. 14