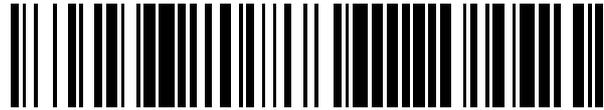


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 503 667**

51 Int. Cl.:

**G09G 5/24**

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **16.11.2005 E 05110825 (6)**

97 Fecha y número de publicación de la concesión europea: **18.06.2014 EP 1659567**

54 Título: **Gestión de datos de fuentes**

30 Prioridad:

**19.11.2004 GB 0425433**

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

**07.10.2014**

73 Titular/es:

**VESTEL ELEKTRONIK SANAYI VE TICARET A.S.  
(100.0%)**

**Organize Sanayi Bolgesi  
45030 Manisa, TR**

72 Inventor/es:

**HEMINGWAY, PETER**

74 Agente/Representante:

**DE ELZABURU MÁRQUEZ, Alberto**

**ES 2 503 667 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

## Gestión de datos de fuentes

5 La presente invención se refiere a la gestión de datos de fuentes y, particularmente, a un dispositivo de almacenamiento de datos de fuentes y un procedimiento de estructuración de datos de fuentes para su almacenamiento en una memoria. La presente invención se refiere a la gestión de datos de fuentes que es particularmente, pero no exclusivamente, adecuado para su uso con pantallas de menor resolución, tales como pantallas emisoras.

10 Las pantallas de menor resolución se emplean en decodificadores y otros dispositivos electrónicos de consumo que tienen potencia limitada y/o recursos de procesamiento y de memoria limitados que incluyen, pero no se limitan a, teléfonos móviles, PDAs y reproductores MP3. En general, las imágenes de números y letras a ser mostrados en dichas pantallas de menor resolución se almacenan como mapas de bits o mapas de píxeles suavizados, en el que cada mapa de píxeles comprende un primer plano del glifo en combinación con el espacio de fondo en y alrededor del glifo. Convencionalmente, para una fuente particular, todos los caracteres alfanuméricos y la puntuación asociada se almacenan de manera individual en mapas de píxeles suavizados, con el mismo tamaño fijo de anchura. Por lo tanto, el tamaño del mapa de píxeles debe ser suficiente para acomodar el glifo con el asta más ascendente, el glifo con el asta más descendente y el glifo con el carácter más ancho. Esto resulta en que solo parte del área de cualquiera de los mapas de píxeles está cubierta por el primer plano del glifo. Esto a su vez significa que, para los componentes electrónicos que tienen recursos limitados, los recursos críticos, tales como espacio de memoria y potencia de procesamiento, se usan para almacenar y procesar píxeles de fondo sin información en cada mapa de píxeles.

20 Una alternativa conocida para almacenar caracteres alfanuméricos son las fuentes de contorno. Sin embargo, el número de instrucciones necesarias para representar los caracteres individuales es tal que las fuentes de contorno generalmente requieren mucha más memoria que los mapas de píxeles, especialmente para los tipos de letra no occidentales. Además, antes de que pueda mostrarse un número o una letra almacenada como datos de fuentes de contorno, primero los datos de fuentes de contorno deben ser convertidos ("rasterizados"). Esto significa que las fuentes de contorno demandan también muchos más recursos de procesamiento que las fuentes de mapas de píxeles. Por consiguiente, las fuentes de contorno no son adecuadas para su uso con dispositivos electrónicos que tienen recursos de procesamiento y/o de memoria limitados.

30 En el documento US 4.481.602, se describe un procedimiento de compresión de fuentes en el que la altura de los caracteres individuales almacenados en la memoria de fuentes varía entre los caracteres. Esto permite cierta reducción en el tamaño de la memoria de fuentes, pero para que el procedimiento funcione la anchura de cada carácter se fija a una anchura máxima para el conjunto de caracteres.

35 Un formato de fuente para la visualización de caracteres basados en texto que incluye una cabecera simplificada se describe en el documento US 5.771.034. La información de cabecera para cada uno de los glifos incluye información sobre la anchura en bytes y la altura en byte del glifo.

Por lo tanto, un objeto de la presente invención es proporcionar una gestión de datos de fuentes mejorada que use menos memoria para el almacenamiento de las fuentes en comparación con los formatos de datos de fuentes existentes.

40 Por consiguiente, en un primer aspecto, la presente invención proporciona un dispositivo de almacenamiento de datos de fuentes según la reivindicación 1. El dispositivo de almacenamiento de datos de fuentes comprende una memoria en la que se almacena una pluralidad de mapas de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a un tipo de letra y cada mapa de píxeles está acompañado de datos representativos de las dimensiones primera y segunda del mapa de píxeles, en el que al menos algunos de los mapas de píxeles tienen dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles en la misma familia de caracteres.

45 Con la presente invención, debido a que los caracteres están representados por mapas de píxeles en los que las dimensiones de los mapas de píxeles pueden variar, el tamaño de los datos de los mapas de píxeles para cada carácter individual puede ser minimizado y, de esta manera, la invención ofrece una estructura de datos más eficiente para almacenar los datos de fuentes en términos de demanda de espacio de memoria. Además, debido a que la presente invención reduce el número de píxeles a ser procesados para cada carácter, esto a su vez reduce la carga del procesador y tiene el potencial de aumentar la velocidad de procesamiento.

50 En un segundo aspecto, la presente invención proporciona un producto de software de datos de fuentes según la reivindicación 15. El producto de software de datos de fuentes comprende una pluralidad de mapas de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a un tipo

de letra y cada mapa de píxeles está acompañado de datos representativos de las dimensiones primera y segunda del mapa de píxeles, en el que al menos algunos de los mapas de píxeles tienen las dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles en la misma familia de caracteres.

5 En un tercer aspecto, la presente invención proporciona un procedimiento de estructuración de datos de fuentes para su almacenamiento en una memoria según la reivindicación 26. La estructura de los datos de fuentes comprende una pluralidad de mapas de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a la fuente y en el que cada mapa de píxeles está acompañado por datos representativos de las dimensiones primera y segunda del mapa de píxeles, en el que al menos algunos de los mapas de píxeles tienen dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles en la misma familia de caracteres.

Idealmente, cada mapa de píxeles incluye datos de cabecera y los datos representativos de las dimensiones primera y segunda del mapa de píxeles forman parte de los datos de cabecera y las dimensiones primera y segunda de los mapas de píxeles se definen con respecto a una altura y una anchura.

15 En una realización, las dimensiones primera y segunda de los mapas de píxeles están representadas por separado en los datos de cabecera por un código de altura y/o un código de anchura. Además, uno o ambos de entre el código altura y el código de anchura pueden ser representativos, respectivamente, de la diferencia de la altura del área de la imagen y la anchura del área de la imagen con respecto a una altura predeterminada y una anchura predeterminada.

20 En una realización alternativa, las dimensiones primera y segunda de los mapas de píxeles están representadas conjuntamente en los datos de cabecera por un código de área de imagen y en el que el dispositivo de almacenamiento incluye además una tabla de consulta en la que se almacenan una pluralidad de combinaciones de dimensiones primera y segunda y sus respectivos códigos de área de imagen.

25 En un cuarto aspecto, la presente invención proporciona un dispositivo electrónico que comprende un procesador; un dispositivo de almacenamiento de datos de fuentes según se ha descrito anteriormente; una pantalla; y un convertidor para acceder a los mapas de píxeles relacionados con uno o más caracteres en el dispositivo de almacenamiento de datos de fuentes y comunicar los datos de imagen para los uno o más caracteres a la pantalla.

30 El archivo de datos de fuentes de la presente invención, en el almacenamiento de caracteres como mapas de píxeles de tamaño variable, es especialmente adecuado para su uso como una fuente incorporada en los sistemas con recursos de procesamiento y/o de memoria limitados, por ejemplo, teléfonos móviles, PDAs, reproductores de MP3 y decodificadores.

Las realizaciones de la presente invención se describirán ahora, a modo de ejemplo, con referencia a los dibujos adjuntos, en los que:

35 La Figura 1 ilustra un mapa de bits convencional de la letra "A" asignada a una matriz rectangular fija de 5x9 píxeles;

La Figura 2 ilustra un mapa de bits convencional de la letra "a" asignada a la misma matriz rectangular fija de 5x9 píxeles;

La Figura 3 muestra un mapa de bits convencional de la letra 'g' asignada a la misma matriz rectangular fija de 5x9 píxeles;

40 La Figura 4 ilustra un mapa de bits de la letra "a" asignada a una matriz rectangular de 3x5 píxeles, según la presente invención;

La Figura 5 ilustra la información de cabecera para dos mapas de píxeles suavizados sucesivos de diferentes tamaños, según la presente invención;

45 La Figura 6 ilustra la información de cabecera para dos mapas de píxeles suavizados sucesivos de diferentes tamaños que usan valores de diferencia, según la presente invención; y

La Figura 7 es un diagrama de bloques funcional de una implementación de la gestión de datos de fuentes según la presente invención adecuada para su implementación en un dispositivo electrónico, tal como un decodificador.

50 La referencia en la presente memoria a datos de fuentes debería entenderse como una referencia a la familia de caracteres, incluyendo caracteres alfanuméricos y puntuación asociada, que tienen un tipo de letra común y a sus equivalentes no occidentales. Además, en el contexto de la presente memoria, la referencia a un mapa de píxeles

pretende hacer referencia a datos de imagen en la forma de una matriz de píxeles en la que cada píxel está representado por uno o más bits.

5 La Figura 1 ilustra una "A" mayúscula asignada a una matriz de píxeles rectangular convencional, o recuadro, de 5x9 píxeles. Debido al tamaño del carácter en mayúscula, tal como puede verse, una gran proporción de los píxeles del recuadro 5x9 se usan para representar este carácter. Sin embargo, en la Figura 2 se ilustra una "a" minúscula asignada al mismo tamaño de recuadro, 5x9 píxeles. Para este carácter, hay varias filas y columnas de píxeles vacíos. De manera similar, en la Figura 3 se ilustra una "g" minúscula y, una vez más, hay varias filas y columnas de píxeles vacíos.

10 Por otra parte, con el procedimiento de la presente invención, los caracteres de un tipo de letra particular son asignados a matrices de píxeles rectangulares o recuadros que tienen dimensiones primera y segunda variables, en este caso altura y anchura. Idealmente, las dimensiones primera y segunda de cada recuadro se eligen de manera que no exista ninguna fila o columna de píxeles vacíos, aunque para ciertos caracteres pueda ser necesario que haya una o más filas o columnas de píxeles vacíos. De esta manera, según la presente invención, la letra "A" mayúscula de la Figura 1 se asignaría a un recuadro de 5x7 píxeles y la letra "a" minúscula de la Figura 2 se asignaría a un recuadro de sólo 3x5 píxeles, tal como se ilustra en la Figura 4.

15 Con el procedimiento de la presente invención, los mapas de píxeles suavizados para cada carácter se almacenan en un archivo de datos de fuentes, junto con la información de cabecera respectiva. Los datos para cada carácter pueden estar marcados para indicar la posición en el archivo del siguiente mapa de píxeles o el archivo de datos de fuentes puede estar organizado de manera que un carácter individual puede ser accedido rápidamente con respecto a una lista de desplazamientos con respecto al comienzo del archivo de datos. La información de cabecera para cada mapa de píxeles incluye datos acerca de cualquiera y preferiblemente ambas de las dimensiones primera y segunda, concretamente, la altura y la anchura, en términos del número de píxeles, del mapa de píxeles. En una realización, en la información de cabecera, el número total de bits antes e incluido un bit de complemento, por ejemplo, "0,0,0,1" identifica la altura del mapa de píxeles y, de manera similar, el siguiente número total de bits antes e incluido un bit de complemento identifica la anchura. De esta manera, "0,0,0,0,1,0,0,1" (o "1,1,1,1,0,1,1,0") es una representación de la información de cabecera necesaria para el mapa de píxeles ilustrado en la Figura 4, que tiene una altura de 5 píxeles y una anchura de 3 píxeles. Este procedimiento de identificación del tamaño de cada mapa de píxeles es eficiente cuando el número de píxeles para cada mapa de píxeles es pequeño. Sin embargo, hay procedimientos alternativos para representar el área de imagen del mapa de píxeles en la cabecera que ofrecen reducciones adicionales en el número de bits. De hecho, en el caso de un mapa de píxeles que tiene 1 píxel de anchura y 1 píxel de alto, la información de tamaño en la cabecera para el mapa de píxeles puede consistir en un único bit. En la Figura 5 se ilustra la información de cabecera en el archivo de datos de fuentes para dos mapas de píxeles secuenciales. La información de cabecera para el primer mapa de píxeles indica que el mapa de píxeles tiene 3 píxeles de altura y 4 píxeles de anchura. La información de cabecera para el segundo mapa de píxeles indica que el segundo mapa de píxeles tiene 5 píxeles de altura y 1 píxel de anchura.

20 Al reducir el tamaño de cada mapa de píxeles al número mínimo de píxeles necesarios para visualizar el glifo y añadiendo información de cabecera a cada mapa de píxeles, que identifica su altura y anchura, el espacio de memoria requerido para el archivo de datos de fuentes se reduce en comparación con los archivos de datos de fuentes convencionales basados en mapas de bits de tamaño de anchura fijo. El ejemplo anterior no maximiza la eficiencia con la que la información de cabecera es rellenada con información del tamaño. Idealmente, la distribución de tamaños de los mapas de píxeles puede ser analizada de manera que a los tamaños más comunes para la altura y la anchura de los mapas de píxeles se les asigne valores que usan el menor número de bits, por ejemplo, usando codificación de Huffman.

25 A continuación, se expone un ejemplo de una manera de reducir el número de bits necesarios para identificar el tamaño de un mapa de píxeles individual. Para muchos tipos de letra, las distribuciones de anchura y altura de los mapas de píxeles estarán sesgadas hacia un máximo, es decir, la anchura/altura media será mayor que la mitad de la anchura/altura máxima. Por ejemplo, si todos los caracteres se asignan a matrices de píxeles que tienen tamaños no mayores de 5x7 píxeles, la anchura y la altura media de los mapas de píxeles será generalmente mayores que 3 píxeles. Para los mapas de bits 5x9 ilustrados en las Figuras 1, 2 y 3, la anchura y la altura media es respectivamente 3,67 y 5,67. Por lo tanto, generalmente, la anchura y la altura de cada mapa de píxeles pueden ser almacenadas de manera más eficiente como valores de diferencia en comparación con máximos conocidos.

30 La Figura 6 muestra la información de cabecera para dos mapas de píxeles sucesivos, donde la altura y la anchura de cada mapa de píxeles se identifican como un valor de diferencia con respecto a un número máximo predeterminado de píxeles. De esta manera, el número total de bits antes e incluyendo un bit de complemento (debe realizarse un cálculo de diferencia contando desde cero) indica lo pequeña que es la altura o la anchura del mapa de píxeles con respecto al número máximo de píxeles. El ejemplo de la Figura 5 supone una altura máxima predeterminada para cada mapa de píxeles de 7 píxeles y una anchura máxima predeterminada para cada mapa

de píxeles de 5 píxeles. De esta manera, la información de cabecera para el primer mapa de píxeles en la Figura 6 identifica una diferencia de 1 píxel con respecto al máximo para la altura del mapa de píxeles y una diferencia de 2 píxeles con respecto al máximo para la anchura. Por consiguiente, el primer mapa de píxeles tiene una altura de 6 píxeles (7 menos 1) y una anchura de 3 píxeles (5 menos 2). Para el segundo mapa de píxeles, no hay ningún bit vacío que precede al primer "1" en la información de cabecera, lo que indica una diferencia de cero para la altura del mapa de píxeles y "0,1" indica una diferencia de 1 para la anchura del mapa de píxeles. Por lo tanto, el segundo mapa de píxeles tiene una altura de 7 píxeles y una anchura de 4 píxeles. De esta manera, puede verse que usando valores de diferencia, en lugar de un recuento directo de bits en la información de cabecera, para indicar el tamaño de cada mapa de píxeles, puede reducirse el número de bits en la información de cabecera.

Por supuesto, será evidente que cuando se usan valores de diferencia, la altura y la anchura máximas predeterminadas para todos los mapas de píxeles pueden ser almacenadas por separado en el archivo de datos de mapa de bits de fuentes o pueden ser almacenadas en cualquier otro lugar al que el procesador acceda siempre que se acceda al archivo de datos de mapa de bits de fuentes. Por ejemplo, aunque puede almacenarse sólo un tipo de letra de un tamaño determinado (por ejemplo, 10 pt Times Roman) que tiene una única altura máxima y una única anchura máxima, se necesitarían tres estructuras de datos de fuentes independientes para representar los caracteres normales, en negrita y en cursiva. Debido a que el sistema usa las mismas altura máxima y anchura máxima fijas para cada archivo de datos de fuentes, no es necesario almacenar estos valores individualmente en los archivos de datos de fuentes.

Con el procedimiento descrito hasta el momento, los mapas de píxeles (o recuadros de caracteres) tienen anchuras y alturas variables (con respecto a los valores reales o de diferencias) que se incluyen en la información de cabecera para cada mapa de píxeles. En una realización alternativa de la presente invención, se asignan anchuras y alturas seleccionadas a los mapas de píxeles de entre una diversidad de anchuras y alturas predeterminadas. En aras de la simplicidad, se hará referencia ahora solamente al uso de alturas predeterminadas. Sin embargo, el mismo procedimiento puede ser usado igualmente para identificar las anchuras de los mapas de píxeles.

En la implementación más simple, a todos los mapas de píxeles se les asigna una de entre una pluralidad de primeras dimensiones predeterminadas, por ejemplo alturas: H1 o H2 etc. Por ejemplo, al carácter en mayúscula de la Figura 1 se le puede asignar la altura H1 (7 píxeles) y al carácter en minúscula de la Figura 2 se le puede asignar la altura H2 (5 píxeles). La información de cabecera para un mapa de píxeles individual requeriría entonces un único bit para indicar si ese mapa de píxeles tiene altura H1 o H2. Es decir, el primer bit de tamaño en la información de cabecera se establece al valor "0" si la altura del mapa de píxeles es H1, o al valor "1" si la altura es H2 o viceversa. Si la segunda dimensión, en este caso la anchura, del mapa de píxeles está limitado también a una selección de dos tamaños alternativos, entonces el tamaño de un mapa de píxeles individuales puede ser identificado plenamente usando sólo dos bits. En esta realización alternativa, la altura de cada mapa de píxeles se incluye en la información de cabecera como un número fijo de bits, en el que los bits de cada mapa de píxeles identifican una selección de una altura y una anchura de entre una pluralidad de tamaños predeterminados seleccionables. Al almacenar el tamaño de cada mapa de píxeles de esta manera, el número de bits necesarios para identificar el tamaño del mapa de píxeles en la información de cabecera se reduce adicionalmente cuando el número de alturas y anchuras seleccionables es pequeño. Sin embargo, debido a que cada uno de los mapas de píxeles está limitado a alturas/anchuras predeterminadas, la cantidad de memoria necesaria para almacenar todos los mapas de píxeles puede ser mayor que para los ejemplos proporcionados anteriormente debido a que ciertos mapas de píxeles pueden incluir una o más filas de píxeles vacíos, en los que la altura y/o la anchura real del carácter es menor que los tamaños seleccionables predeterminados. Por ejemplo, si las alturas seleccionables predeterminadas son 5 y 7 píxeles, un carácter que tiene una altura de sólo 3 píxeles tendrá dos filas de píxeles vacíos en el mapa de píxeles.

Aquí también, las alturas y anchuras predeterminadas seleccionables de los mapas de píxeles pueden ser almacenadas con el archivo de datos de fuentes o pueden ser almacenadas en otro lugar en el sistema, por ejemplo, en una tabla de consulta. Puede emplearse un sistema similar, pero con un único código que representa, de manera conjunta, tanto la altura como la anchura de la zona de la imagen del mapa de píxeles.

Preferiblemente, la información de cabecera para cada mapa de píxeles incluye también un desplazamiento x, y (la posición del carácter en relación a un carácter contiguo). Una vez más, en áreas de la simplicidad, se hará referencia sólo al desplazamiento de los mapas de píxeles en la dirección (y) vertical. Sin embargo, el desplazamiento x (dirección horizontal) puede ser incluido de manera similar y puede incluir, de manera alternativa o adicional, el ajuste de la separación entre pares de caracteres de manera que la separación varía para diferentes pares de caracteres.

Tal como se ha indicado anteriormente, los mapas de bits de tamaño fijo convencionales requerían acomodar los caracteres con el asta más descendente (letras g, y, j, p y q) y los caracteres con el asta más ascendente (letras l,

t, f, h, k y b), así como el los caracteres con las excursiones más altas (por ejemplo, Á y Ü) y el resto de las letras se disponían dentro del espacio del tamaño fijo de mapa de bits de manera que ser alineasen apropiadamente. De esta manera, las letras tales como la letra "a" se posicionaban centralmente dentro del recuadro de carácter con filas de píxeles vacíos por encima y por debajo del glifo. De esta manera la alineación relativa de los caracteres individuales estaba incluida en los mapas de bits de tamaño fijo individuales.

Al adoptar los mapas de píxeles de tamaño variable, la posición relativa de un carácter individual desaparece del contenido del mapa de píxeles y, en su lugar, se incluye preferentemente en la información de cabecera para cada mapa de píxeles. De esta manera, además de la altura y la anchura del mapa de píxeles, la información de cabecera puede incluir además valores de desplazamiento vertical y horizontal con respecto a las líneas de referencia respectivas. En la dirección vertical, la línea de referencia puede ser la línea base común para todas las letras que no tienen asta descendente, en el que el desplazamiento vertical identifica el número de píxeles, por debajo de la línea de referencia, donde debería colocarse la base de un carácter. De esta manera, la letra 'g' puede tener un desplazamiento vertical de 2, mientras que todas las letras no tienen asta descendente tendrán un desplazamiento vertical de cero. Por supuesto, de manera alternativa, la línea de referencia puede identificar la posición más baja de la base de cualquier carácter individual en la que las letras que tienen un asta descendente, tal como la letra 'g', tendrían un desplazamiento nulo y todas las letras que no tienen asta descendente tendrían, por ejemplo, un desplazamiento de 2 píxeles por encima de la línea de referencia.

Los valores de desplazamiento pueden ser incluidos en la información de cabecera como valores reales o pueden ser almacenados como una selección de entre una pluralidad de posiciones de desplazamiento predeterminadas seleccionables. En este último caso, la posición de desplazamiento vertical de cada mapa de píxeles puede ser incluida en la información de cabecera como una de entre dos posiciones seleccionables predeterminadas: P1 y P2. Por ejemplo, a todos los caracteres que no tienen un asta descendente se les asigna la posición P1, mientras que a los caracteres que tienen un asta descendente se les asigna la posición P2. Entonces, los mapas de píxeles que tienen una posición de desplazamiento P1 se visualizan 2 píxeles más arriba que los mapas de píxeles que tienen una posición de desplazamiento P2. Esto permite que la posición de desplazamiento sea identificada en la cabecera por medio de un único bit de información. Por supuesto, pueden emplearse reglas alternativas para identificar los valores de desplazamiento en la información de cabecera, usando un número mínimo de bits.

De manera similar a las alturas y anchuras predeterminadas seleccionables, las posiciones de desplazamiento predeterminadas (P1 y P2) o su valor de diferencia (por ejemplo, P1-P2) pueden ser almacenados en el archivo de datos de fuentes o en otra parte del sistema.

En la Figura 7 se ilustra esquemáticamente una implementación de la gestión de datos de fuentes descrita en la presente memoria, adecuada para su uso en un decodificador u otro dispositivo electrónico con recursos limitados. La pantalla 10 de un dispositivo electrónico, tal como un decodificador, recibe los datos que debe mostrar desde un procesador 11. A su vez, el procesador 11 recibe instrucciones para construir una imagen a ser mostrada a través de una interfaz 12 de pantalla de datos de entrada. En el caso de un decodificador, la interfaz 12 de pantalla de datos de entrada está adaptada para recibir instrucciones desde un procesador local (no ilustrado) y/o desde un receptor de satélite o por cable (no ilustrado) con relación a la imagen a mostrar.

En respuesta a las instrucciones recibidas a través de la interfaz 12 de datos de pantalla de entrada, el procesador 11 recupera desde el almacenamiento 13 de datos de fuentes los caracteres necesarios para visualizar la imagen requerida. En la Figura 7, los mapas de píxeles están almacenados en una memoria 14 separada de una primera tabla 15 de consulta para los tamaños de mapas de píxeles y una segunda tabla 16 de consulta para los datos de desplazamiento. Por lo tanto, en este ejemplo, la información de cabecera para cada mapa de píxeles almacenado en el almacenamiento de datos de fuentes incluye códigos representativos de las dimensiones del mapa de píxeles y el desplazamiento a aplicar. Entonces, estos códigos pueden ser usados para localizar en las tablas 15, 16 las dimensiones reales de los mapas de píxeles y el desplazamiento para cada mapa de píxeles.

A continuación, los datos de imagen son convertidos por un convertidor 17 a una forma adecuada para su salida a la pantalla 10. Por lo tanto, el convertidor 17 interpreta las dimensiones de los mapas de píxeles y el desplazamiento para cada mapa de píxeles para preparar una tabla intermedia, por ejemplo, que representa la imagen de todos los caracteres que se visualizarán en términos de una sucesión de líneas. A continuación, esta información es comunicada por el procesador 11 a la pantalla 10, línea por línea según es generada por el convertidor o sólo una vez convertida toda la imagen.

El funcionamiento general del procesador 11 es controlado por una unidad 18 de control, cuya función principal es la de secuenciar el funcionamiento de los diversos elementos del procesador. Se apreciará que, aunque la unidad 18 de control se ilustra como una entidad funcional distinta, en la práctica sus funciones están incluidas frecuentemente en otros elementos funcionales del procesador 11.

Al almacenar la información de posición relativa en términos de valores de desplazamiento, con la presente

invención, ya no hay ninguna necesidad de que los mapas de píxeles incluyan filas de píxeles vacíos simplemente para conseguir una alineación correcta con relación a otros caracteres y puede conseguirse una reducción adicional en la cantidad de memoria necesaria para el archivo de datos de fuentes.

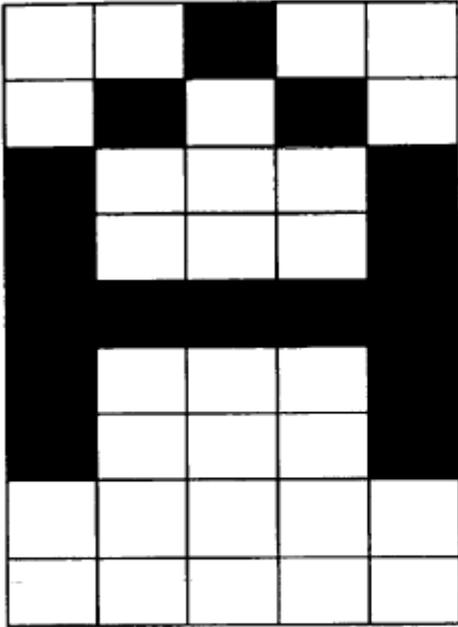
5 Con el procedimiento de grabación y almacenamiento de datos de fuentes descrito anteriormente, los datos de mapa de bits de fuentes pueden ser almacenados de manera que se utiliza menos memoria. En particular, los mapas de bits de caracteres individuales se almacenan con un mínimo de filas o columnas de píxeles en blanco. Se consiguen ahorros adicionales en espacio de memoria gracias a la manera en la que se comunica el tamaño y la posición de desplazamiento en la información de cabecera para cada mapa de bits.

10 Por supuesto, la presente invención puede ser usada también en conjunción con procedimientos conocidos de compresión de fuentes. En particular, un archivo de datos de fuentes que contiene mapas de píxeles de tamaños variables puede ser sometido a una codificación por grupos de longitud variable ("Run-Lenth Encoding") para comprimir adicionalmente el tamaño del archivo.

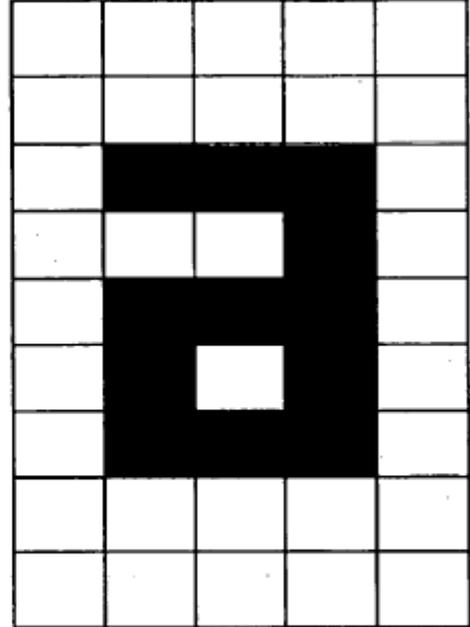
**REIVINDICACIONES**

1. Un dispositivo de almacenamiento de datos de fuentes que comprende una memoria (13) en la que se almacena una pluralidad de mapas (14) de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a un tipo de letra y cada mapa de píxeles está acompañado de datos representativos de las dimensiones primera y segunda del mapa (15) de píxeles, en el que al menos algunos de los mapas de píxeles tienen dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles, en el que cada mapa de píxeles incluye datos de cabecera y los datos representativos de las dimensiones primera y segunda del mapa de píxeles forman parte de los datos de cabecera, en el que las dimensiones primera y segunda de los mapas de píxeles se definen con respecto a una altura y una anchura del área de imagen de los mapas de píxeles y están representadas por separado en los datos de cabecera por un código de altura y un código de anchura, **caracterizado por que** uno o ambos de entre el código de altura y el código de anchura son representativos, respectivamente, de la diferencia entre la altura del área de la imagen y un valor de altura predeterminado y/o la anchura del área de la imagen y un valor de anchura predeterminado en el que el valor de altura predeterminado y el valor de anchura predeterminado son respectivamente la altura máxima y la anchura máxima para cualquier carácter que pertenece a una familia de fuentes de los caracteres.
2. Dispositivo de almacenamiento de datos de fuentes según la reivindicación 1, en el que cada mapa de píxeles tiene asociado con el mismo datos (16) de desplazamiento vertical que son representativos de una posición desplazada vertical para el carácter respectivo, en el que los datos de desplazamiento vertical comprenden un único bit de información.
3. Dispositivo de almacenamiento de datos de fuentes según cualquiera de las reivindicaciones 1 o 2, en el que cada mapa de píxeles tiene asociado con el mismo datos de desplazamiento horizontal representativos de una posición desplazada horizontal para el carácter respectivo, en el que los datos de desplazamiento horizontal incluyen la provisión de una separación entre el carácter respectivo y un carácter contiguo en una cadena de caracteres.
4. Dispositivo de almacenamiento de datos de fuentes según las reivindicaciones 2 y 3, en el que cada mapa de píxeles incluye datos de cabecera en los que los datos de desplazamiento vertical y los datos de desplazamiento horizontal forman parte de los datos de cabecera.
5. Dispositivo de almacenamiento de datos de fuentes según una cualquiera de las reivindicaciones anteriores, en el que cada mapa de píxeles consiste en un mapa de bits.
6. Dispositivo de almacenamiento de datos de fuentes según una cualquiera de las reivindicaciones anteriores, en el que la pluralidad de mapas de píxeles se comprimen y almacenan en la memoria con una codificación por grupos de longitud variable.
7. Dispositivo de almacenamiento de datos de fuentes según una cualquiera de las reivindicaciones anteriores, en el que la memoria (13) del dispositivo de almacenamiento es un medio de almacenamiento permanente o re-escribible.
8. Un dispositivo electrónico que comprende un procesador, un dispositivo de almacenamiento de datos de fuentes según una cualquiera de las reivindicaciones anteriores, una pantalla (10) y un convertidor (17) para acceder a los mapas de píxeles relacionados con uno o más caracteres del dispositivo (13) de almacenamiento de datos de fuentes y comunicar los datos de imagen para los uno o más caracteres a la pantalla.
9. Dispositivo electrónico según la reivindicación 8, en el que el dispositivo electrónico es un decodificador.
10. Un producto de software de datos de fuentes que comprende una pluralidad de mapas de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a un tipo de letra y cada mapa de píxeles está acompañado de datos representativos de las dimensiones primera y segunda del mapa de píxeles, en el que al menos algunos de los mapas de píxeles tienen dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles, en el que cada mapa de píxeles incluye datos de cabecera y los datos representativos de las dimensiones primera y segunda del mapa de píxeles forman parte de los datos de cabecera, en el que las dimensiones primera y segunda de los mapas de píxeles se definen con respecto a una altura y una anchura del área de imagen de los mapas de píxeles y están representadas por separado en los datos de cabecera por un código de altura y un código de anchura, **caracterizado por que** uno o ambos de entre el código de altura y el código de anchura son representativos, respectivamente, de la diferencia entre la altura del área de la imagen y un valor de altura predeterminado y/o la anchura del área de la imagen y un valor de anchura predeterminado, en el que el valor de altura predeterminado y el valor de anchura predeterminado son, respectivamente, la altura y la anchura máximas para cualquier carácter que pertenece a la familia de fuentes de los caracteres.

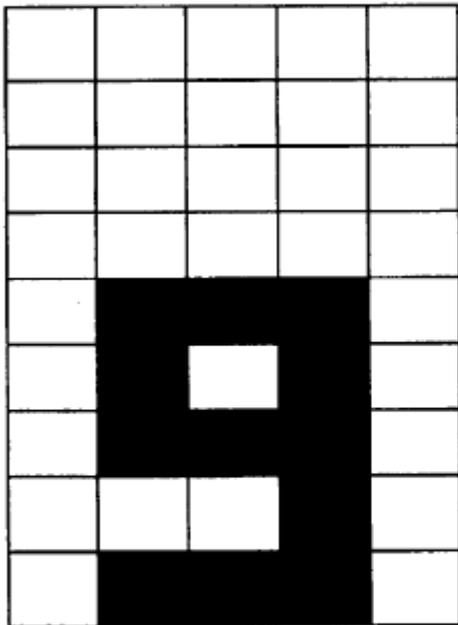
11. Producto de software de datos de fuentes según la reivindicación 10, en el que cada mapa de píxeles tiene asociado con el mismo datos de desplazamiento vertical representativos de una posición desplazada vertical para el carácter respectivo, en el que los datos de desplazamiento vertical comprenden un único bit de información.
- 5 12. Producto de software de datos de fuentes según cualquiera de las reivindicaciones 10 o 11, en el que cada mapa de píxeles tiene asociado con el mismo datos de desplazamiento horizontal representativos de una posición desplazada horizontal para el carácter respectivo, en el que los datos de desplazamiento horizontal incluyen la provisión de una separación entre el carácter respectivo y un carácter contiguo en una cadena de caracteres.
- 10 13. Producto de software de datos de fuentes según las reivindicaciones 11 y 12, en el que cada mapa de píxeles incluye datos de cabecera en los que los datos de desplazamiento vertical y los datos de desplazamiento horizontal forman parte de los datos de cabecera.
14. Producto de software de datos de fuentes según una cualquiera de las reivindicaciones 10 a 13, en el que cada mapa de píxeles consiste en un mapa de bits.
- 15 15. Un producto de software de datos fuente según una cualquiera de las reivindicaciones 10 a 14, en el que la pluralidad de mapas de píxeles se comprimen y almacenan en la memoria con una codificación por grupos de longitud variable.
- 20 16. Un procedimiento de estructuración de datos de fuentes para su almacenamiento en una memoria, en el que los datos de fuentes comprenden una pluralidad de mapas de píxeles, en el que cada mapa de píxeles representa un carácter respectivo de la familia de caracteres que pertenecen a la fuente y cada mapa de píxeles está acompañado de datos representativos de las dimensiones primera y segunda del mapa de píxeles, en el que al menos algunos de los mapas de píxeles tienen dimensiones primera y segunda que difieren de las dimensiones primera y/o segunda de otros mapas de píxeles, en el que cada mapa de píxeles incluye datos de cabecera y los datos representativos de las dimensiones primera y segunda del mapa de píxeles forman parte de los datos de cabecera, en el que las dimensiones primera y segunda de los mapas de píxeles se definen con respecto a una altura y una anchura y están representados por separado en los datos de cabecera por un código de altura y código de anchura, **caracterizado por que** uno o ambos de entre el código de altura y el código de anchura son representativos, respectivamente, de la diferencia entre la altura del área de la imagen y un valor de altura predeterminado y/o la anchura del área de la imagen y un valor de anchura predeterminado, en el que el valor de altura predeterminado y el valor de anchura predeterminado son, respectivamente, la máxima altura y la máxima anchura para cualquier carácter que pertenece a la familia de fuentes de caracteres.
- 25 17. Procedimiento de estructuración de datos de fuentes según la reivindicación 16, en el que cada mapa de píxeles tiene asociado con el mismo datos de desplazamiento vertical representativos de una posición desplazada vertical para el carácter respectivo, en el que los datos de desplazamiento vertical comprenden un único bit de información.
- 30 18. Procedimiento de estructuración de datos de fuentes según cualquiera de las reivindicaciones 16 o 17, en el que cada mapa de píxeles tiene asociado con el mismo datos de desplazamiento horizontal representativos de una posición desplazada horizontal para el carácter respectivo, en el que los datos de desplazamiento horizontal incluyen la provisión de la separación entre el carácter respectivo y un carácter contiguo en una cadena de caracteres.
- 35 19. Procedimiento de estructuración de datos de fuentes según las reivindicaciones 17 y 18, en el que cada mapa de píxeles incluye datos de cabecera y los datos de desplazamiento vertical y los datos de desplazamiento horizontal forman parte de los datos de cabecera.
- 40 20. Procedimiento de estructuración de datos de fuentes según una cualquiera de las reivindicaciones 16 a 19, en el que cada mapa de píxeles consiste en un mapa de bits.
- 45 21. Procedimiento de estructuración de datos de fuentes según cualquiera de las reivindicaciones 16 a 20, en el que los datos de cabecera incluyen además datos para la alineación del carácter con otros caracteres cuando es visualizado.
- 50 22. Procedimiento de estructuración de datos de fuentes según una cualquiera de las reivindicaciones 16 a 21, en el que la pluralidad de mapas de píxeles se comprimen y se almacenan en la memoria con una codificación por grupos de longitud variable.



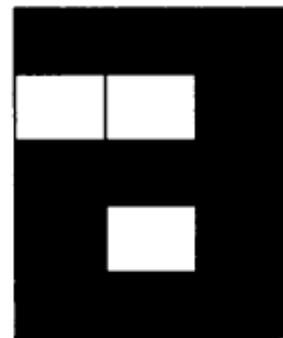
**Fig. 1**



**Fig. 2**



**Fig. 3**



**Fig. 4**



Fig. 5



Fig. 6

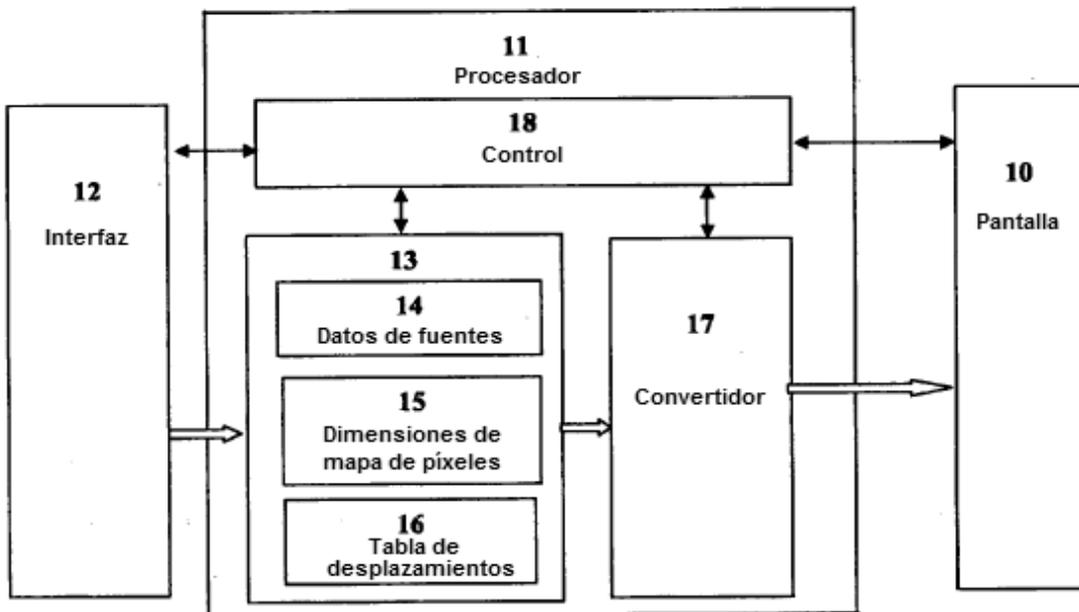


Fig. 7