

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 565 338**

51 Int. Cl.:

G06F 3/14 (2006.01)

G06F 3/033 (2013.01)

G06F 17/00 (2006.01)

G06F 3/048 (2013.01)

G06F 3/0481 (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **01.07.2010 E 10794757 (4)**

97 Fecha y número de publicación de la concesión europea: **16.12.2015 EP 2449456**

54 Título: **Puntero aumentado basándose en vectores**

30 Prioridad:

02.07.2009 US 222709 P

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

04.04.2016

73 Titular/es:

**FREEDOM SCIENTIFIC, INC. (100.0%)
11800 31st Court North
St. Petersburg, FL 33716, US**

72 Inventor/es:

**STRINGER, ANTHONY BOWMAN y
VOORHEES, GARALD LEE**

74 Agente/Representante:

DÍAZ DE BUSTAMANTE TERMINEL, Isidro

ES 2 565 338 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

Puntero aumentado basándose en vectores.

5 REFERENCIA CRUZADA A SOLICITUD RELACIONADA

Esta solicitud reivindica prioridad de la solicitud de patente provisional de Estados Unidos actualmente pendiente de tramitación N.º de serie 61/222.709 presentada el 2 de julio 2009 titulada "Magnified Pointing Indicia Method".

CAMPO DE LA INVENCION

10 Esta invención se refiere al aumento de interfaces de pantalla de ordenador, y más específicamente, al aumento de signos indicadores en su interior.

ANTECEDENTES DE LA INVENCION

15 Los individuos con visión deficiente a menudo requieren aumento de las interfaces de pantalla de ordenador para distinguir texto e imágenes. Los sistemas de aumento pueden estar incorporados en el propio sistema operativo o pueden comprender productos de terceras partes con muchas características, tales como los comercializados con la marca MAGIC fabricados por Freedom Scientific, Inc., con sede en St. Petersburg, Florida.

20 Una deficiencia en el actual estado de la técnica es la de la modificación a escala de signos indicadores tales como un cursor o signo de inserción de pequeñas a grandes dimensiones. Por ejemplo, todos los punteros de ratón tienen un tamaño mínimo discreto y uno máximo común. El tamaño mínimo es un múltiplo de la medida del sistema para el puntero de ratón. Generalmente, esta medida es un cuadrado de 32x32 píxeles en sistemas operativos comercializados por Microsoft Corporation de Redmond, Washington con las marcas WINDOWS 7, VISTA y XP. Sin embargo, es habitual que un puntero de ratón ocupe solamente un cuadrado de 16x16, justificado en la parte superior izquierda, dentro de esta medida.

25 Cuando la pantalla se aumenta a un nivel alto, tal como 16 veces, el puntero de ratón se pixela y distrae al usuario. Puede aplicarse diversas técnicas de suavizado para reducir la pixelación, pero dichos procesos consumen ciclos de la CPU y siguen produciendo resultados imperfectos.

30 Otra deficiencia del estado actual de la técnica es el sobreaumento de los signos indicadores. Los usuarios con visión deficiente a menudo están limitados en el aumento que pueden imponer, dado que los signos indicadores son aumentados al compás con toda la interfaz gráfica del usuario. Por lo tanto, a niveles de aumento elevados (es decir, que superan las 16 veces) los signos indicadores se vuelven tan grandes que ocultan gran parte de la propia interfaz gráfica del usuario.

35 La patente de Estados Unidos N.º 2006/095867 A1 desvela un método y sistema que presenta automáticamente un cursor más visible cuando se produce un evento. Además, la patente de Estados Unidos N.º 2005/204295 A1 desvela un producto de software informático para optimizar una interfaz gráfica del usuario para uso por un individuo con visión deficiente. Además, la patente de Estados Unidos N.º 2002/180792 A1 desvela la combinación de la funcionalidad de múltiples controles de texto en una interfaz gráfica del usuario. Finalmente, la patente de GB N.º 2 245 130 A desvela arquitectura de bus local de visualización y método de comunicaciones para rasterizar la pantalla.

40 La patente de Estados Unidos N.º 7111 254 B1 desvela un sistema para sustituir la imagen de un cursor en relación con la visualización del contenido de una página web. Además, la patente de Estados Unidos N.º 2008/195979 A1 desvela una interfaz para ocultar un cursor o mostrar un cursor a un controlador de video y para hacer que información del cursor se guarde en un almacenamiento de datos y se recupere del almacenamiento de datos.

RESUMEN DE LA INVENCION

La invención se expone en la reivindicación 1.

45 Una realización de la presente invención proporciona un método para proporcionar un signo indicador aumentado que incluye las etapas de monitorizar la mensajería de la interfaz de programación de aplicaciones, interceptar una llamada para un actual identificador de puntero del sistema (tal como su descriptor "handle"), acceder a una colección almacenada de formas vectoriales predefinidas, seleccionar una forma vectorial predefinida de la colección, la forma vectorial seleccionada correlacionada con el identificador de puntero del sistema, modificar a escala la forma vectorial a un nivel de aumento, rasterizar la forma vectorial modificada a escala, y visualizar la forma rasterizada. En lugar, o además de, interceptar de forma pasiva una llamada para un identificador de puntero del sistema actual, el sistema operativo es sondeado activamente en cuanto a sus actuales ajustes y el identificador de puntero del sistema actual se resuelve.

Para cargar punteros de ratón en un entorno de Microsoft Windows en el que puede usarse la invención, aplicaciones solicitan a la función API *LoadCursor*:

```
HCURSOR LoadCursor(HINSTANCE hInstance,LPCTSTR LPCTSTR
lpCursorName);
```

- 5 Si *hInstance* es NULL, entonces *lpCursorName* contiene un identificador que representa uno de los cursores del sistema estándar. Estos valores pueden incluir:

Valor	Significado
IDC_APPSTARTING MAKEINTRESOURCE(32650)	Flecha estándar y reloj de arena pequeño
IDC_ARROW MAKEINTRESOURCE(32512)	Flecha estándar
IDC_CROSS MAKEINTRESOURCE(32515)	Punto de mira
IDC_HAND MAKEINTRESOURCE(32649)	Mano
IDC_HELP MAKEINTRESOURCE(32651)	Fecha y signo de interrogación
IDC_IBEAM MAKEINTRESOURCE(32513)	Viga en I
IDC_ICON MAKEINTRESOURCE(32641)	Obsoleto para aplicaciones comercializadas con versión 4.0 o posterior.
IDC_NO MAKEINTRESOURCE(32648)	Círculo cortado
IDC_SIZE MAKEINTRESOURCE(32640)	Obsoleto para aplicaciones comercializadas con versión 4.0 o posterior. Usar IDC_SIZEALL.
IDC_SIZEALL MAKEINTRESOURCE(32646)	Flecha cuatro puntas que señalan al norte, sur, este y oeste
IDC_SIZENESW MAKEINTRESOURCE(32643)	Flecha de dos puntas que señalan al noreste y sudoeste
IDC_SIZENS MAKEINTRESOURCE(32645)	Flecha de dos puntas que señalan al norte y al sur
IDC_SIZENWSE	Flecha de dos puntas que señalan al noroeste y
MAKEINTRESOURCE(32642)	sudeste
IDC_SIZEWE MAKEINTRESOURCE(32644)	Flecha de dos puntas que señalan al oeste y al este
IDC_UPARROW MAKEINTRESOURCE(32516)	Flecha vertical
IDC_WAIT MAKEINTRESOURCE(32514)	Reloj de arena

- 10 Si *hInstance* no es NULL, entonces *lpCursorName* contiene un identificador numérico o de secuencia que representa una imagen gráfica contenida en los recursos de la aplicación. En una realización, la presente invención crea un diccionario de programación que cartografía el identificador devuelto por *lpCursorName* a una forma vectorial correlacionada (que puede ser de diversos formatos incluyendo, aunque sin limitarse a CGM, SVG, ODG, EPS, PDF, SWF, WMF, EMF y XPS). El diccionario y la correlación con el identificador del puntero pueden ser realizados mediante una serie de técnicas de programación que incluyen, aunque sin limitarse a, un archivo INI, archivo XML, archivo CSV, motor de base de datos, servicio web o similares. La identificación del cursor a imagen del vector puede ser de diversas relaciones que incluyen muchos a uno o uno a uno.

- 15 *LoadCursor* devuelve un descriptor numérico al cursor que se cargó. La aplicación en primer plano selecciona el cursor activo llamando a la función *SetCursor*. Esta función toma uno de los descriptores HCURSOR creados previamente por *LoadCursor*. Interceptando ambas de estas funciones, es posible seguir el rastro del cursor activo que está siendo usado por una aplicación. Interceptando la función *SetForegroundWindow*, es posible determinar qué aplicación está en primer plano y, por lo tanto, rastrear el cursor de qué aplicación está activo en todo el sistema.
- 20

La capacidad de modificar a escala suavemente un puntero de ratón u otro elemento gráfico se mejora

enormemente usando imágenes basadas en vectores para crear las imágenes rasterizadas almacenadas en memoria caché iniciales que se usan en última instancia para la renderización final (dibujo en la pantalla). La razón para esto es que las formas y elementos de las imágenes basadas en vectores tal como se describe mediante una fórmula, en lugar de mediante un mapa de bits. Actualmente, todos los punteros de ratón se describen mediante mapas de bits. Los mapas de bits se vuelven inherentemente “en forma de bloques” y “en forma de escalones” cuando son aumentados. Y aunque hay tratamientos post-modificación a escala que pueden aplicarse para “suavizar” artificialmente un mapa de bits modificado a escala, ninguno produce imágenes que sean tan definidas y con bordes tan nítidos como las imágenes producidas a partir de una imagen vectorial modificada a escala.

Un archivo de imagen basada en vectores tal como un archivo *.WMF puede importarse y correlacionarse con uno o más descriptores o identificadores de puntero de ratón. Para facilitar la eficiencia y la sensibilidad, formas rasterizadas previamente a diversos niveles de aumento pueden almacenarse y recuperarse mediante un caché sin tener que modificar a escala y rasterizar las formas de nuevo. El almacenamiento de formas se realiza normalmente en dispositivos de almacenamiento disponibles que pueden incluir, aunque no se limitan a, discos duros magnéticos, discos en estado sólido y/o RAM.

Una realización alternativa de la invención incluye etapas para establecer un punto de convergencia para un aumento máximo de signos indicadores y establecer un nivel de aumento del escritorio seleccionable por el usuario, con lo que los niveles de aumento de los signos indicadores y de aumento del escritorio están sincronizados hasta que se alcanza el punto de convergencia. Pasado ese punto, el nivel de aumento del escritorio puede incrementarse pero el aumento de los signos indicadores permanece estático. Cuando el nivel de aumento del escritorio cae por debajo del punto de convergencia, entonces el aumento de los signos indicadores se resincroniza con el nivel de aumento del escritorio. El punto de convergencia para desacoplar y reacoplar el aumento de puntero-GUI (interfaz gráfica del usuario) puede basarse en una serie de factores que incluyen, aunque sin limitarse a, solapamiento del puntero del espacio disponible del escritorio, nivel de aumento de la GUI, el tamaño de los propios signos indicadores, ajustes específicos de la aplicación informática, ajustes de eventos (es decir, escritura en una celda de una hoja de cálculo), eventos en teclado y/o ratón (es decir, movimiento de la rueda del ratón) o similares.

En una realización de la invención, el punto de convergencia es un aumento de aproximadamente 16 veces. En otra realización de la invención, el aumento máximo de los signos indicadores constituye aproximadamente el veinte por ciento del área total del escritorio según se calcula para la dimensión mayor de la pantalla. Por ejemplo, un escritorio que es de 680x480 píxeles tiene un área de 307.200 píxeles cuadrados. Un puntero aumentado hasta utilizar 61.440 píxeles constituiría el veinte por ciento del área total del escritorio y se consideraría, por lo tanto, un “valor techo” para el tamaño del puntero. En otra realización más de la invención, el punto de convergencia se calcula basándose en un porcentaje de área de la pantalla de acuerdo con la dimensión menor de la pantalla. Usando la dimensión menor de la pantalla, el valor de convergencia alcanzar un comportamiento razonable cuando se realiza en pantallas muy desproporcionadas. Por ejemplo, un monitor que presenta una resolución nativa de 1360x768 píxeles produce una dimensión mayor que es casi 2x la menor. En condiciones como estas, es preferible usar la dimensión menor, dado que eso garantiza que el valor de convergencia no superará nunca ninguna dimensión de la pantalla, a menos que sea solicitado explícitamente por el usuario. En el ejemplo anterior, un valor de convergencia del 60%, basándose en la dimensión mayor, sería, de hecho, de 816 píxeles verticales y superaría el total de la dimensión menor de la pantalla. En otra realización más de la invención, el punto de convergencia y el aumento máximo de los signos indicadores son específicos de la aplicación. Por ejemplo, en una aplicación de hoja de cálculo, puede ser necesario establecer un punto de convergencia menor, con lo que el signo indicador es relativamente más pequeño para permitir a un usuario ver celdas de la hoja de cálculo más fácilmente sin solapamiento. Como alternativa, un programa de pruebas que solicita al usuario que seleccione respuestas de elección múltiple podría tener un punto de convergencia más elevado. La convergencia específica de aplicación puede preseleccionarse y/o ser definida por el usuario de acuerdo con las necesidades del individuo con visión deficiente.

Otra realización de la invención proporciona un nivel de aumento del escritorio seleccionable por el usuario que es independiente de un nivel de aumento de los signos indicadores seleccionable por el usuario. En esta realización de la invención, el usuario invoca ratón, teclado y/u otros comandos periféricos para modificar el nivel de aumento de los signos indicadores independientemente del nivel de aumento de la GUI. Por ejemplo, mantener pulsadas una o más teclas en un teclado mientras se mueve la rueda de un ratón acercaría o alejaría la GUI, mientras que mantener pulsadas teclas diferentes mientras se mueve la rueda del ratón acercaría o alejaría solamente los signos indicadores. Un primer comando de pulsado de tecla puede sincronizar el aumento de la GUI y los signos indicadores. Un segundo comando de pulsado de tecla puede desincronizar el aumento de la GUI y los signos indicadores.

La invención puede materializarse en uno o más medios legibles por ordenador que tienen instrucciones ejecutables por ordenador para realizar un método de ejecución de un programa informático en un ordenador, funcionando el ordenador regido por un sistema operativo, incluyendo el método emitir instrucciones desde el programa informático que incluyen identificar un identificador de puntero del sistema actual a partir de la interfaz de programación de aplicaciones del sistema operativo, acceder a una colección almacenada de formas vectoriales predefinidas, seleccionar una forma vectorial predefinida de la colección, la forma vectorial seleccionada correlacionada con el identificador de puntero del sistema actual, modificar a escala la forma vectorial a un nivel de aumento, rasterizar la

forma vectorial modificada a escala, y visualizar la forma rasterizada. Las instrucciones almacenadas en los medios legibles por ordenador ejecutan como una aplicación, distinta de un sistema operativo en el que se ejecuta e interactúa a un nivel controlador de la visualización para modificar la salida de la pantalla. Como alternativa, los medios legibles por ordenador pueden incluir un sistema operativo, o actualización del mismo, para conseguir la novedosa funcionalidad de la presente invención.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

Para una comprensión más completa de la invención, debería hacerse referencia a la siguiente descripción detallada, tomada en relación con los dibujos adjuntos, en los que:

- 10 La figura 1 es una vista conceptual de una realización de la invención que muestra un usuario con visión deficiente viendo una pantalla a un aumento de 1 vez.
- La figura 2 es una vista conceptual de una realización de la invención que muestra un usuario con visión deficiente viendo una pantalla a un aumento de 2 veces.
- La figura 3 es una vista conceptual de una realización de la invención que muestra un usuario con visión deficiente viendo una pantalla a un aumento de 4 veces.
- 15 Las figuras 4-5 son vistas conceptuales de una realización de la invención que muestran un usuario con visión deficiente viendo una pantalla a un aumento de 16 veces.
- La figura 6 es un gráfico que muestra un punto de convergencia para tamaño máximo de signos indicadores a un aumento de 16 veces.
- La figura 7 es una interfaz gráfica del usuario para personalizar un puntero de ratón.
- 20 La figura 8 es una interfaz gráfica del usuario para importar archivos basados en vectores y asignarlos como signos indicadores.
- La figura 9 es una vista esquemática de una realización de la invención.
- La figura 10 es una vista esquemática de una realización de la invención.
- 25 La figura 11 es una vista esquemática de una realización de la invención que muestra el sondeo periódico del sistema operativo para obtener los ajustes de signos indicadores actuales.
- La figura 12 es una vista esquemática de una realización de la invención que muestra intercepción de *dvrSetPointerShape* entre el sistema operativo y el controlador del dispositivo de visualización para obtener los ajustes de signos indicadores actuales.

DESCRIPCIÓN DETALLADA DE LA REALIZACIÓN PREFERIDA

- 30 Volviendo ahora a la figura 1, un usuario con visión deficiente 10 manipula las vistas en el monitor 20 mediante el ratón 30 y el teclado 40. El signo indicador 50 se muestra como un cursor de flecha para fines ejemplares pero podría ser también un signo de inserción. Un cursor es la imagen gráfica que representa el ratón y sus movimientos en un plano de coordenadas cartesianas. Puede asumir muchas formas incluyendo, aunque sin limitarse a, una flecha para señalar, una mano, un reloj de arena o un selector de texto en forma de I. El signo de inserción, por otro
- 35 lado, es el signo parpadeante que se usa para introducir texto. Cuando el usuario con visión deficiente 10 escribe en el teclado 40, aparece la letra en el signo de inserción, y el signo de inserción se mueve hacia delante 1 espacio. Para ver los niveles de aumento relativos, en la figura 1 se proporciona visualización de texto 60 y visualización de gráficos 70.
- 40 La figura 1 muestra la GUI, signos indicadores 50, visualización de texto 60 y visualización de gráficos 70 sin aumentar. En la figura 2, se muestran los signos indicadores 50, la visualización de texto 60 y la visualización de gráficos 70, conceptualmente, a un nivel de aumento de 2 veces. En la figura 3, se muestran signos indicadores 50, visualización de texto 60 y visualización de gráficos 70, conceptualmente, a un nivel de aumento de 4 veces. En la figura 4, se muestran signos indicadores 50, visualización de texto 60 y visualización de gráficos 70, conceptualmente, a un nivel de aumento de 16 veces. A este nivel, el signo indicador 50 se solapa
- 45 aproximadamente un veinte por ciento del área de la GUI. De acuerdo con una realización de la presente invención, el aumento adicional de la GUI y la visualización de texto 60 desacopla el aumento previamente sincronizado de la GUI y los signos indicadores 50. El tamaño de los signos indicadores 50 permanece constante mientras que el aumento de la visualización de texto 60 continúa pasado el aumento de 16 veces.
- 50 En la figura 5, el aumento de los signos indicadores 50 se desacopla del aumento de la GUI. La GUI se aumenta hasta 16 veces mientras que los signos indicadores 50 se aumentan hasta solamente 4 veces. El usuario con visión deficiente 10 puede desacoplar el aumento entre los signos indicadores 50 y la GUI usando el teclado 40, el ratón 30, o una combinación de los mismos. Pueden usarse otros periféricos y sistemas de entrada tales como sistemas

de reconocimiento de voz, para efectuar el evento. El desacoplamiento también puede conseguirse automáticamente y de forma inteligente mediante la invención. Por ejemplo, un umbral predeterminado de inactividad del ratón puede detectarse para incrementar o disminuir gradualmente (según se desee) el aumento de los signos indicadores 50.

5 La inactividad del ratón con datos extrínsecos puede usarse para tomar decisiones de aumento inteligentes. Por ejemplo, si se detecta una gran cantidad de texto en la GUI junto con inactividad del ratón, puede realizarse una presunción de que el usuario está leyendo el texto. En tal caso, un cursor muy aumentado solamente ocultaría la visión del texto. Como alternativa, el movimiento del ratón 30 inmediatamente después de un periodo de inactividad del ratón puede invocar un aumento rápido y escalonado de los signos indicadores 50 de modo que puedan ser localizados fácilmente por el usuario con visión deficiente cuando reanuda el uso de los signos indicadores 50. El aumento escalonado puede usarse para proporcionar un efecto animado. En otros casos, una pluralidad de formas vectoriales modificadas gradualmente puede proporcionar otros efectos de animación tales como movimiento, rotación, formas de onda y similares. Dado que las formas están basadas en vectores, agrandadas hasta el nivel de aumento, rasterizadas y almacenadas en caché, la calidad del gráfico y la eficiencia de ciclado de las imágenes almacenadas en caché proporcionan una ventaja sustancial respecto al actual estado de la técnica.

Otra realización más de la invención incluye la etapa de detectar algorítmicamente movimientos erráticos del ratón 30 por el usuario con visión deficiente 10 después de un periodo de inactividad del ratón 30 que es indicativo de dificultad para encontrar la ubicación del signo indicador 50. El movimiento errático puede incluir, aunque no se limita a, movimiento rápido de un lado a otro y movimiento circular. Sensible a la detección de movimientos erráticos, el signo indicador 50 es aumentado de acuerdo con la invención durante un periodo de tiempo predeterminado y a continuación vuelve a su nivel de aumento preseleccionado. Una ventaja de esta realización respecto a la técnica anterior es que el movimiento errático debe seguir a un periodo predeterminado de inactividad, dado que el movimiento del ratón percibido como "errático" puede ser intencionado si el usuario está inmerso en una aplicación informática que requiere dicho movimiento.

25 La figura 6 ilustra un gráfico de convergencia que muestra cuatro (4) tamaños de puntero diferentes indicados como pequeño (◆), normal (■), grande (▲) y enorme (×). El valor de tamaño del puntero pequeño es 0,5x. El valor de tamaño del puntero normal es 1,0x. El valor de tamaño del puntero grande es 2,0x. El valor de tamaño del puntero enorme es 4,0x. A un aumento de 1 vez de la GUI, los tamaños de puntero pequeño, normal, grande y enorme cubren un área de la GUI de ~0%, 3%, 5% y 10% respectivamente. A medida que el aumento de la GUI se incrementa, la variación de tamaños de puntero produciría experiencias del usuario drásticamente diferentes. Sin embargo, en una realización de la presente invención, estos tamaños de puntero se normalizan a un aumento de la GUI de 16 veces. A medida la GUI es aumentada de 1 vez a 16 veces, el tamaño de puntero pequeño se incrementa sustancialmente más con respecto al tamaño de puntero enorme. También pueden seleccionarse otros niveles de aumento para convergencia.

35 Volviendo a la figura 7, una aplicación de autorización del puntero es anticipada por la presente invención para permitir a usuarios crear formas vectoriales y asignarlas a conjuntos de punteros o punteros individuales. Los conjuntos de punteros contienen colecciones de diferentes imágenes gráficas para cursor puntero que cambian dependiendo de las circunstancias en que se visualizan. Cuando se selecciona texto para editar, un cursor puede cambiar a una barra vertical (viga en I). Cuando se realizan vistas panorámicas en un documento, el cursor puede aparecer como una mano. Cursores de edición de textos pueden aparecer como lapiceros, pinceles y cubos de pintura dependiendo de la función. Cuando se espera a que termine una operación, una imagen de "espera" puede aparecer tal como un reloj de arena o reloj. Otras opciones de indicación incluyen seleccionar aspectos de relleno y línea para color, patrones, transparencia y similares. A las formas se les pueden asignar aplicaciones informáticas específicas. Esto puede conseguirse inspeccionando, la mensajería del sistema operativo para identificar qué aplicación adquiere mayor importancia y aplicar el ajuste de puntero y/o modificación del cursor apropiada según se desee. Dado que el presente método permite la representación rápida y limpia de imágenes nítidas y claras en cualquier dimensión, animar el cursor del puntero es posible. Una secuencia de estilizaciones y/o cambios de dimensión de la forma vectorial puede almacenarse en caché con lo que el cursor puede parecer animado a medida que la secuencia se visualiza. Pueden asignarse parámetros de modificación a escala con la rueda y combinaciones de teclas de acceso directo a formas y/o conjuntos específicos. Por ejemplo, Un usuario puede mantener pulsada una combinación en el teclado mientras hace girar la rueda del ratón en diferentes direcciones para aumentar o reducir a escala el cursor del puntero.

La figura 8 muestra una aplicación de importación para asignar cualquier archivo gráfico basado en vectores a un puntero y/o conjunto de punteros específico. Los formatos de archivo de vector pueden incluir, aunque sin limitarse a, metarchivos de gráficos de ordenador (CGM), gráficos de vector modificables a escala (SVG), gráficos de documento abierto (ODG), descriptor de páginas encapsulado (EPS), formato de documento portátil (PDF), formato web pequeño (SWF), metarchivo de Windows (WMF), metarchivo mejorado de Windows (EMF), y especificación de documento XML (XPS). Otra ventaja de la presente invención es que los es más probable que los usuarios con visión deficiente tengan una gran colección de diseño vectorial preexistente para utilizar como signos indicadores a cualquier nivel de aumento. En contraste, las imágenes rasterizadas son difíciles de incrementar de tamaño mientras se conserva la claridad y la nitidez debido a la pixelación de la imagen en mapa de bits.

5 En la figura 9, se muestra una realización de la invención como el proceso 70. La mensajería API 80 es monitorizada en busca de llamada al descriptor del puntero 90. Se realiza un cotejo 100 contra la base de datos de forma vectorial 110 para emparejar la llamada al descriptor del puntero 90 con una forma preexistente en la base de datos de forma vectorial 110. La base de datos de forma vectorial 110 contiene una o más formas vectoriales almacenadas en la memoria. La base de datos 110 puede ser un directorio de archivos, un archivo estructurado tal como XML, una base de datos completamente relacional o similares. La base de datos de formas vectoriales 110 asocia la llamada al descriptor del puntero 90 con una imagen del vector correlacionada. Por ejemplo, si la llamada al descriptor del puntero 90 solicita una imagen del cursor de una mano, el cotejo 100 determina si existe una imagen del vector para la imagen del cursor de una mano. En caso negativo, entonces una imagen en mapa de bits se aumenta a escala y se suaviza. Sin embargo, si existe una imagen vectorial, entonces se aumenta a escala al nivel de aumento apropiado y a continuación se rasteriza a un mapa de bits. La forma vectorial 120 se recupera y se aplica la función de modificación del tamaño a escala 130 para producir una forma vectorial aumentada 140. La modificación del tamaño a escala se realiza en una imagen vectorial incrementando o reduciendo la longitud y la anchura de la imagen. Dado que la modificación a escala se realiza en una imagen vectorial, no hay pixelación como se observaría en una modificación a escala en mapa de bits. El proceso de rasterización 150 se aplica a la forma vectorial aumentada 140 para producir la forma rasterizada 160 que se visualiza a continuación 170 como los signos indicadores.

20 La figura 10 ilustra una realización alternativa de la invención (indicada como proceso 75) que se adapta a llamadas al descriptor del puntero 90 que no tienen una forma correspondiente en la base de datos de formas vectoriales 110. En el caso de que no se establezca una coincidencia, el signo indicador en mapa de bits por defecto 180 es aumentado a signo indicador en mapa de bits agrandado 190. El procesamiento de suavizado 200 se aplica a signos indicadores en mapa de bits agrandados 190 y a continuación se visualizan 170. En otra realización de la invención, al no encontrar coincidencia para una forma vectorial, el sistema da pie al usuario con visión deficiente 10 a asociar una forma vectorial existente, crear una nueva forma vectorial o importar un archivo basado en vector como un sustituto para signos indicadores en mapa de bits por defecto 180. En otra realización más de la invención, se proporciona una forma vectorial por defecto para llamadas al descriptor del puntero 90 que carecen de una forma correspondiente en base de datos de formas vectoriales 110.

30 La figura 11 ilustra el sondeo periódico 220 del sistema operativo 210 para obtener los ajustes de signos indicadores actuales. Al recuperar estos ajustes, se selecciona la forma vectorial predefinida que se correlaciona con los ajustes de signos sondeados.

35 En la figura 12, un método alternativo de obtención de la forma del puntero incluye la etapa de interceptar el valor de *drvSetPointerShape* 240 entre el sistema operativo 210 y el controlador del dispositivo de visualización 230. El valor de *drvSetPointerShape* 240 se coteja 100 para ver si existe una forma vectorial correlacionada 110 que esté indizada al valor de *drvSetPointerShape* 240. Usar el valor de *drvSetPointerShape* 240 es lo más apropiado para los quince (15) tipos de cursor predefinidos (no obsoletos) disponibles en la especificación de interfaz de programación de aplicaciones de MICROSOFT WINDOWS.

40 Se observará que las ventajas descritas anteriormente, y aquellas que son evidentes a partir de la descripción anterior, se consiguen eficazmente y dado que pueden realizarse ciertos cambios en la construcción anterior sin alejarse del alcance de la invención, se pretende que todos los asuntos contenidos en la descripción anterior o mostrados en los dibujos adjuntos se interpreten como ilustrativos y no en un sentido limitante.

REIVINDICACIONES

- 5 1. Uno o más medios legibles por ordenador que tienen instrucciones ejecutables por ordenador que, en respuesta a su ejecución por un ordenador, hacen que el ordenador aumente un signo indicador (50) realizando operaciones que comprenden:
- identificar un identificador de puntero del sistema actual a partir de una interfaz de programación de aplicaciones del sistema operativo (210);
 - acceder a una colección almacenada de formas vectoriales (120) de signos indicadores (50) predefinidas;
 - 10 seleccionar una forma vectorial predefinida (120) a partir de la colección, la forma vectorial seleccionada (120) correlacionada con el identificador de puntero del sistema actual;
 - modificar a escala la forma vectorial (120) a un nivel de aumento;
 - rasterizar la forma vectorial modificada a escala (120); y
 - renderizar un signo indicador aumentado (50) visualizando la forma rasterizada (160).
- 15 2. Los medios legibles por ordenador de la reivindicación 1, que comprenden además las etapas de importar un archivo de imagen basado en vectores y correlacionar el archivo de imagen basado en vectores con uno o más identificadores únicos de puntero del sistema.
- 20 3. Los medios legibles por ordenador de la reivindicación 1, en los que la etapa de identificar el identificador de puntero del sistema actual comprende las etapas de sondear periódicamente el sistema operativo (210) para sus ajustes de signos indicadores (50) actuales y seleccionar la forma vectorial predefinida (120) que se correlaciona con los ajustes de signos indicadores (50) sondeados.
- 25 4. Los medios legibles por ordenador de la reivindicación 1, en los que la etapa de identificar el identificador de puntero del sistema comprende las etapas de monitorizar la mensajería de la interfaz de programación de aplicaciones del sistema operativo (210), interceptar una llamada para el identificador de puntero del sistema actual, y correlacionar la forma vectorial seleccionada (120) con el identificador de puntero del sistema interceptado.
- 30 5. Los medios legibles por ordenador de la reivindicación 1, en los que la etapa de identificar el identificador de puntero del sistema comprende las etapas de monitorizar comunicaciones al controlador del dispositivo de visualización (230), interceptar una llamada para el identificador de puntero del sistema actual, y correlacionar la forma vectorial seleccionada (120) con el identificador de puntero del sistema interceptado.
- 35 6. Los medios legibles por ordenador de la reivindicación 1, que comprenden además la etapa de almacenar en caché formas rasterizadas (160) para reutilización.
7. Los medios legibles por ordenador de la reivindicación 1, que comprenden además las etapas de:
- 40 establecer un punto de convergencia para un aumento máximo de signos indicadores (50), siendo el punto de convergencia el nivel de aumento por encima del cual el aumento de un signo indicador (50) y el contenido en segundo plano se desacoplan;
 - establecer un nivel de aumento del escritorio;
 - modificar a escala la forma vectorial (120) en sincronización con el nivel de aumento del escritorio hasta el punto de convergencia, con lo que la forma vectorial (120) ya no está aumentada a escala una vez que se alcanza el punto de convergencia, en el que solamente el contenido en segundo plano es aumentado por encima del punto de convergencia mientras que la forma vectorial (120) permanece en el mismo tamaño; y
 - 45 resincronizar la modificación a escala de la forma vectorial (120) y el contenido en segundo plano cuando el nivel

de aumento cae por debajo del punto de convergencia.

8. Los medios legibles por ordenador de la reivindicación 7, en los que el punto de convergencia es un aumento de aproximadamente 16 veces.

5

9. Los medios legibles por ordenador de la reivindicación 7, en los que el aumento máximo de los signos indicadores (50) constituye aproximadamente el veinte por ciento del área total del escritorio según se calcula a partir de la dimensión menor de la pantalla.

10 10. Los medios legibles por ordenador de la reivindicación 7, en los que el punto de convergencia y el aumento máximo de los signos indicadores (50) son específicos de aplicación.

11. Los medios legibles por ordenador de la reivindicación 1, que comprenden además las etapas de:

establecer un nivel de aumento del escritorio; y

15 establecer un nivel de aumento de los signos indicadores (50) seleccionable por el usuario, en el que el nivel de aumento de los signos indicadores (50) es independiente del nivel de aumento del escritorio y la forma vectorial (120) se modifica a escala de acuerdo con el nivel de aumento de los signos indicadores (50).

20 12. Los medios legibles por ordenador de la reivindicación 11, en los que la etapa de establecimiento de un nivel de aumento de los signos indicadores (50) seleccionable por el usuario se realiza en respuesta a un evento de la rueda del ratón.

25 13. Los medios legibles por ordenador de la reivindicación 11, en los que la etapa de establecer un nivel de aumento de los signos indicadores (50) seleccionable por el usuario se realiza en respuesta a un evento de pulsado de una tecla.

14. Los medios legibles por ordenador de la reivindicación 1, que comprenden además las etapas de:

establecer un nivel de aumento de los signos indicadores (50) para una aplicación informática preseleccionada; y

30 modificar a escala la forma vectorial (120) de acuerdo con el nivel de aumento de los signos indicadores (50) en respuesta a una actividad asociada con la aplicación informática preseleccionada.

Fig. 1

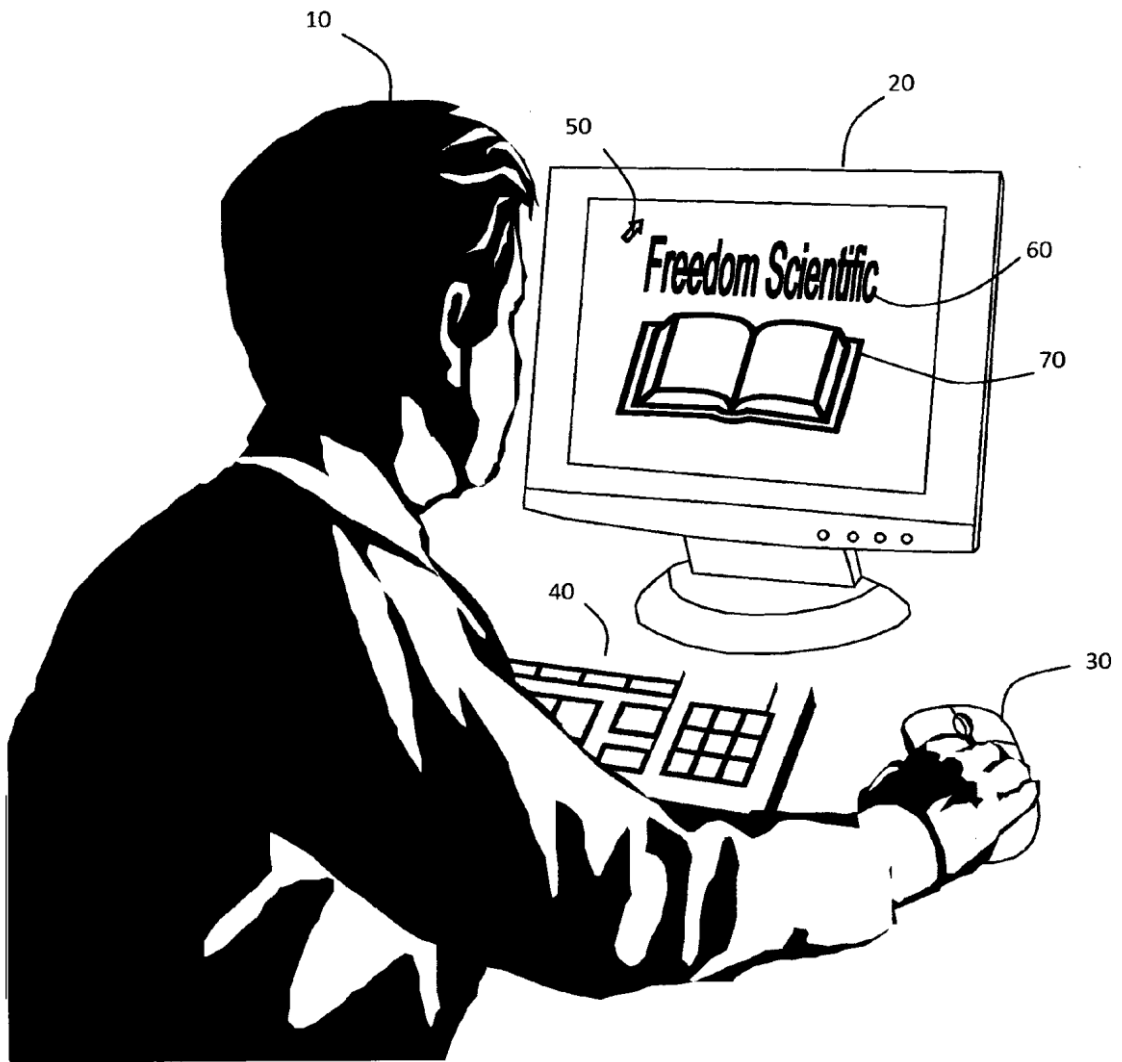


Fig. 2



Fig. 3



Fig. 4



Fig. 5



Fig. 6

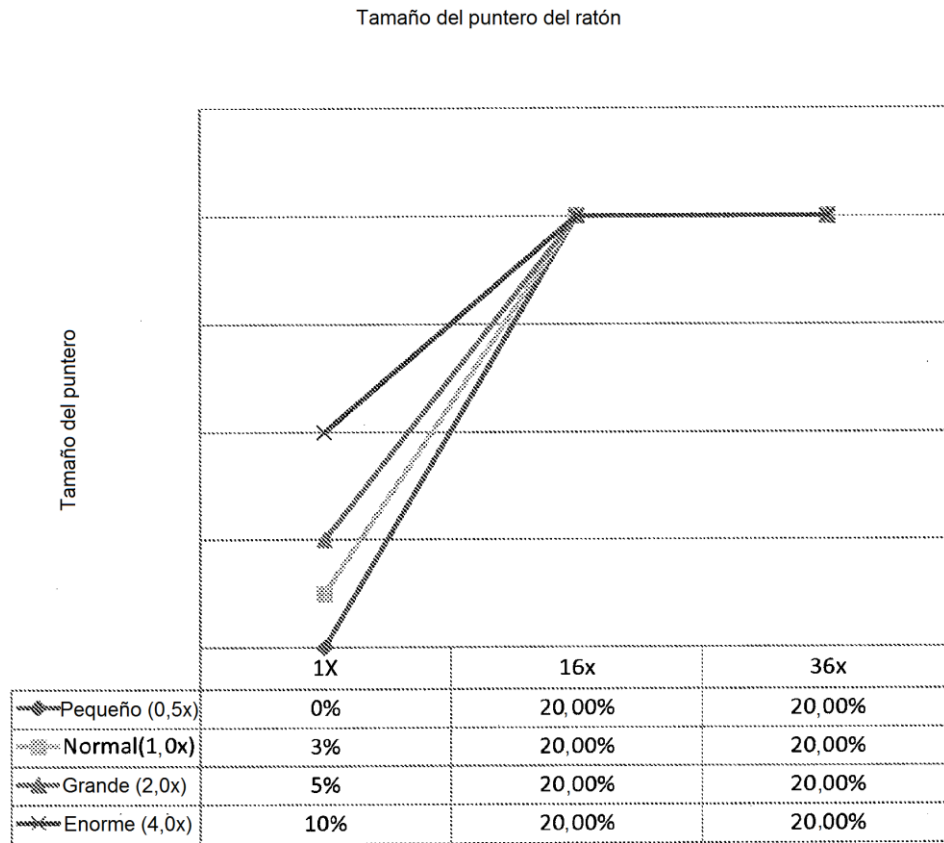


Fig. 7

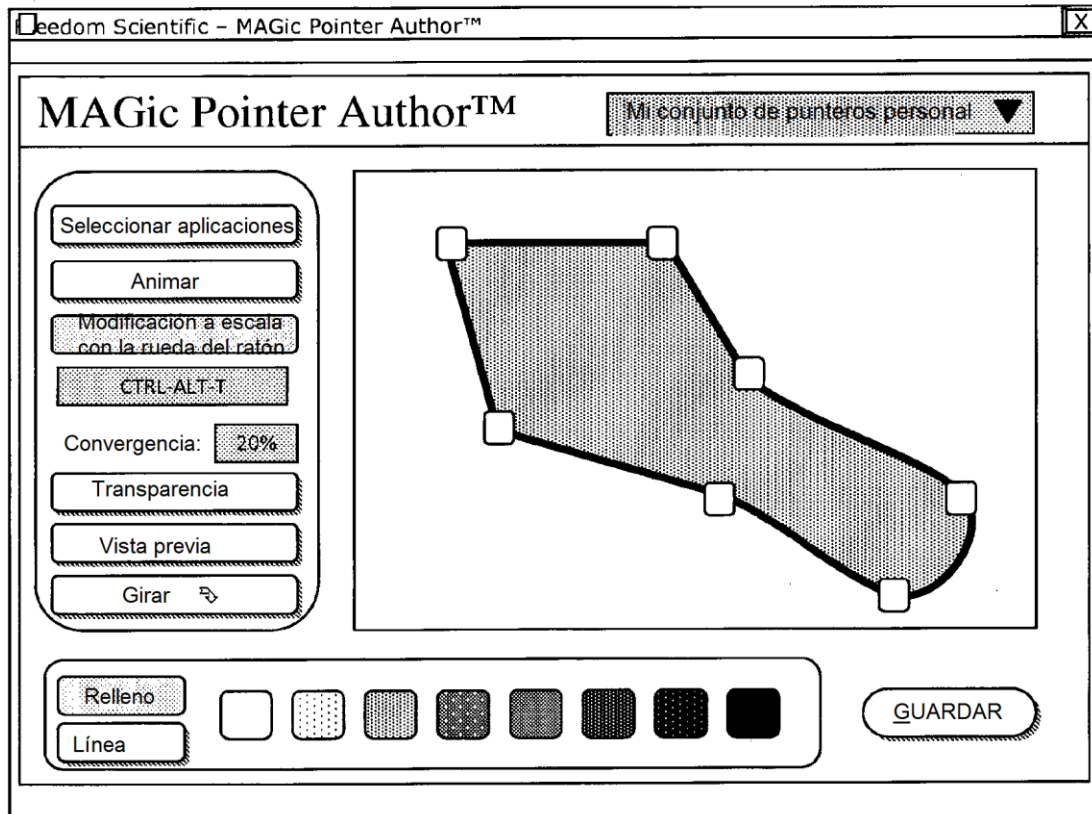


Fig. 8

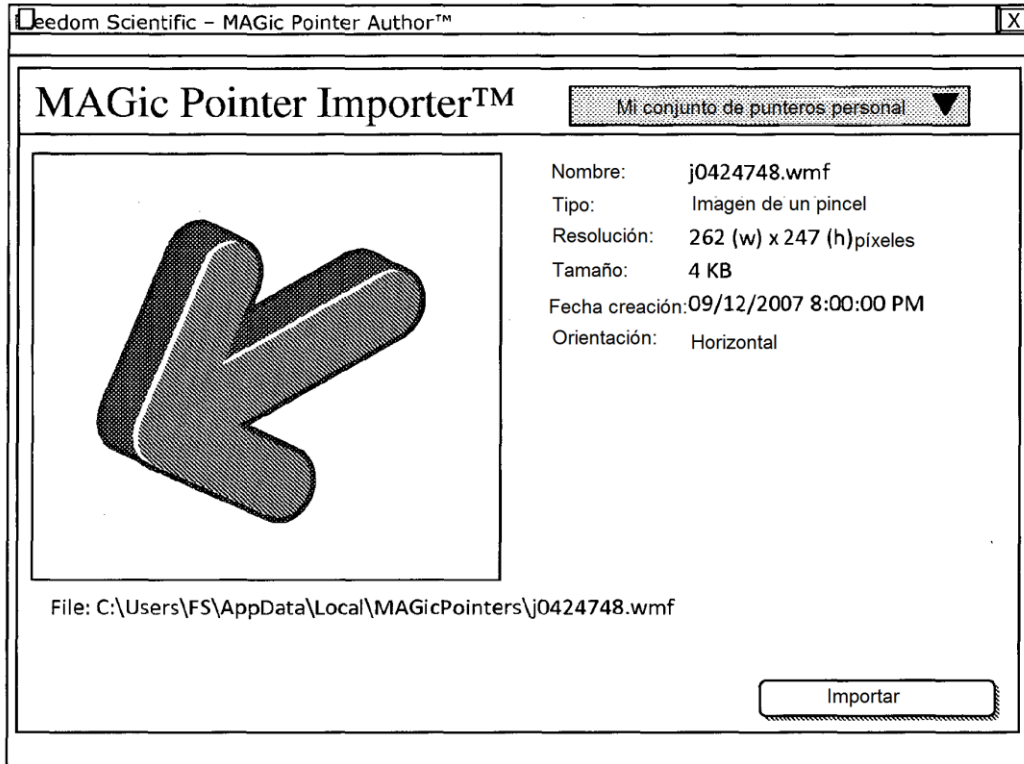


Fig. 9

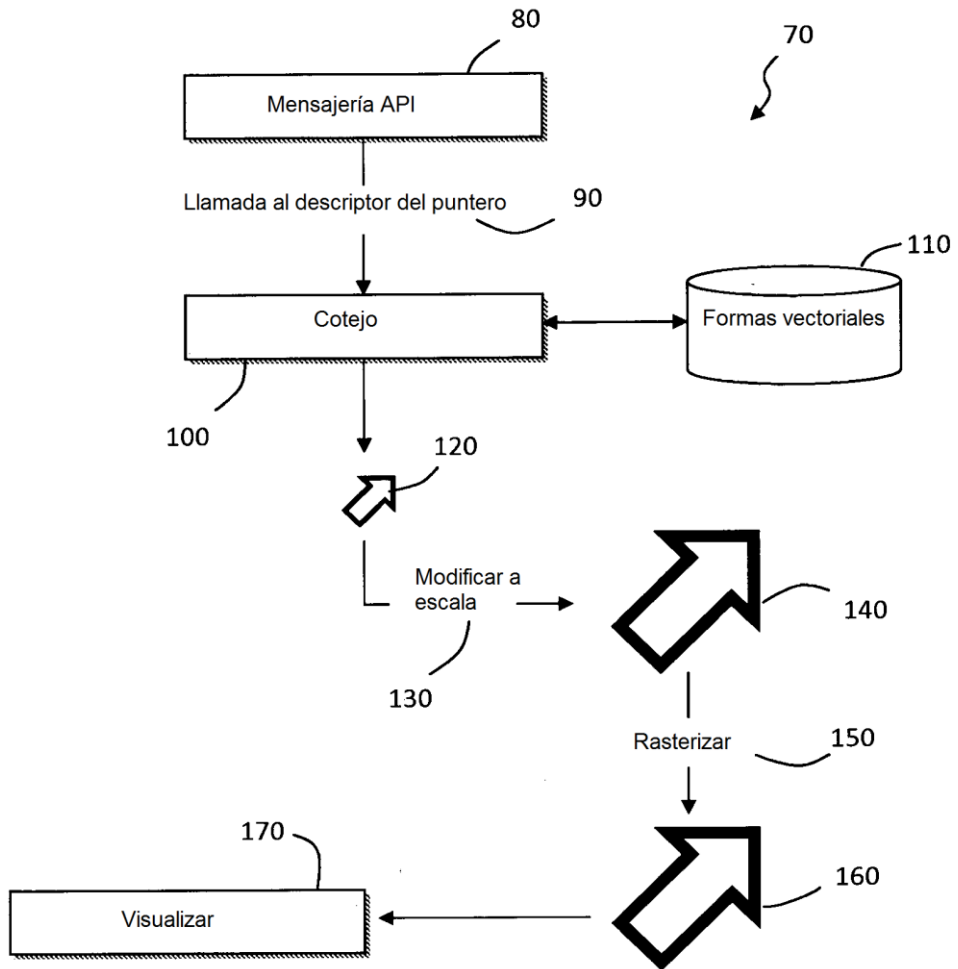


Fig. 10

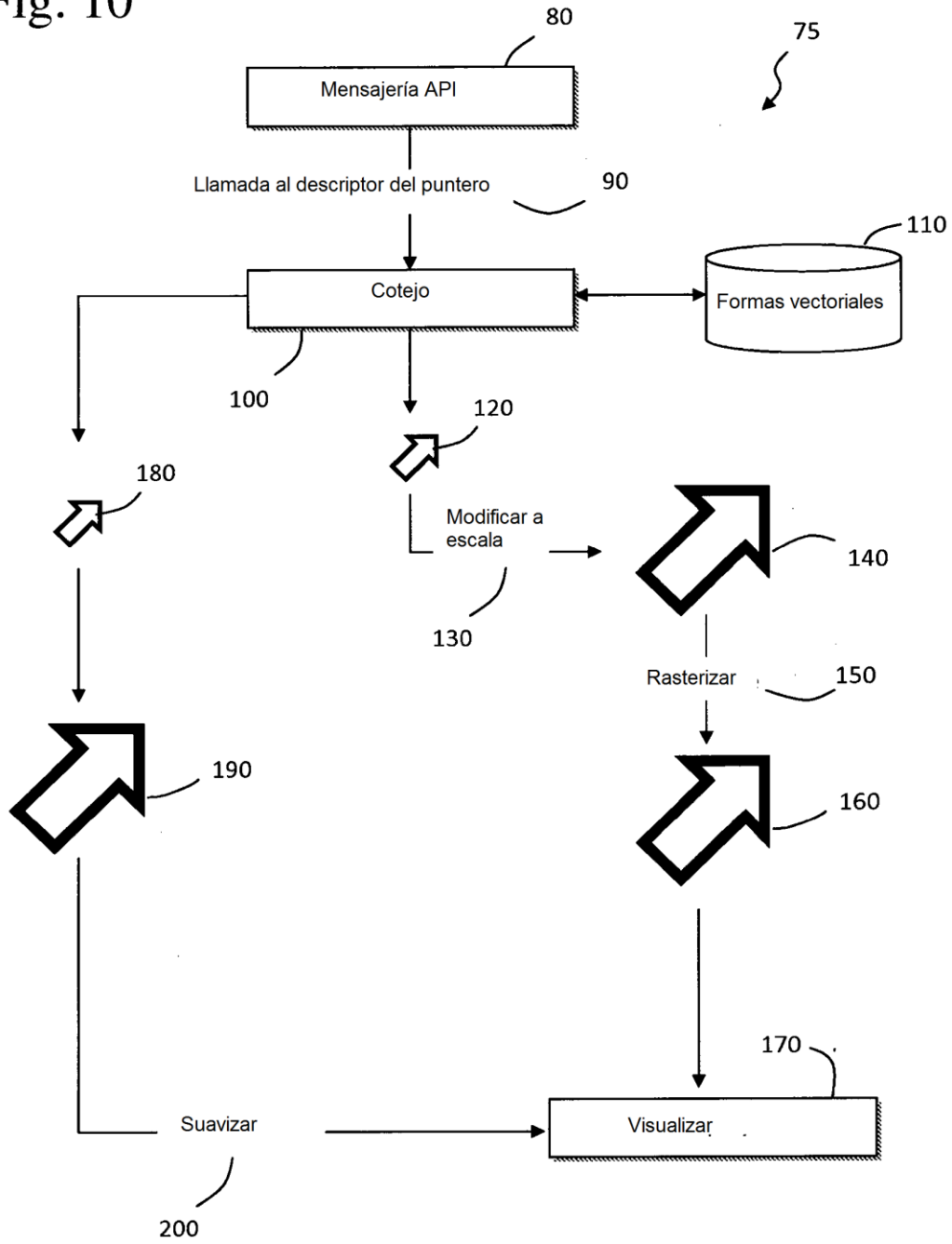


Fig. 11

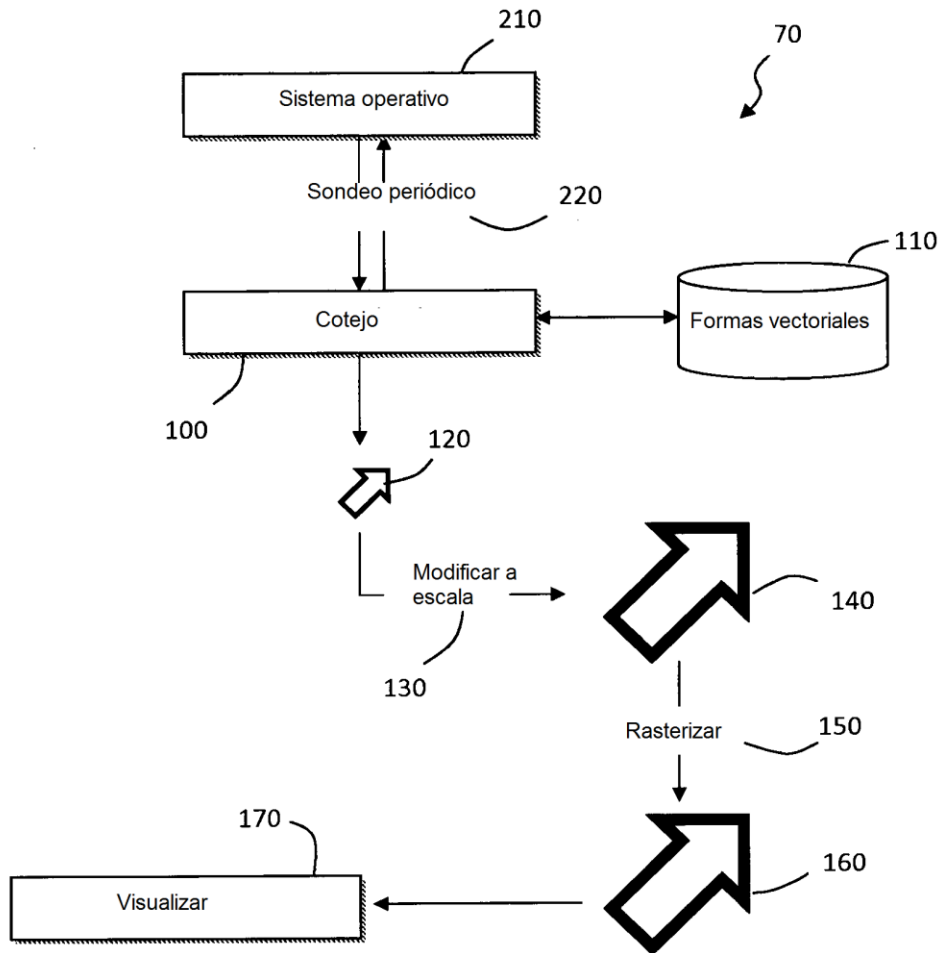


Fig. 12

