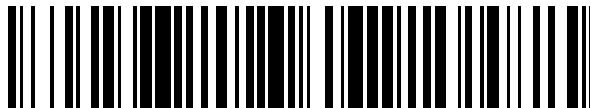


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 690 648**

51 Int. Cl.:

G06F 3/048 (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **13.10.2008 PCT/EP2008/008662**

87 Fecha y número de publicación internacional: **18.06.2009 WO09074185**

96 Fecha de presentación y número de la solicitud europea: **13.10.2008 E 08858967 (6)**

97 Fecha y número de publicación de la concesión europea: **05.09.2018 EP 2235618**

54 Título: **Interfaz de usuario mejorada que tiene efectos físicos realistas**

30 Prioridad:

12.12.2007 US 954845

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

21.11.2018

73 Titular/es:

**NOKIA TECHNOLOGIES OY (100.0%)
Karaportti 3
02610 Espoo, FI**

72 Inventor/es:

**PIKKUJAMSA, KALLE;
LINJAMA, JUKKA;
JUHA, KELA;
PANU, KORPIPÄÄ;
SANNA, KALLIO y
TAPANI, RANTAKOKKO**

74 Agente/Representante:

VALLEJO LÓPEZ, Juan Pedro

ES 2 690 648 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

Interfaz de usuario mejorada que tiene efectos físicos realistas

- 5 La presente solicitud se refiere a un dispositivo y un método para proporcionar un acceso rápido a la información almacenada en o accesible a través de un dispositivo, y en particular a un dispositivo y un método para proporcionar un acceso rápido a los datos almacenados en o accesibles a través de un dispositivo mediante el uso de órdenes de golpecito.

10 **Antecedentes de la invención**

- El documento US2007/0178974 (D1) divulga un sistema de juego que incluye una carcasa que va a ser sujeta por un jugador. De acuerdo con el documento D1, la carcasa incorpora un sensor de aceleración de eje XY para detectar una aceleración en una dirección de eje X y de eje Y y un conmutador de contacto de eje Z para detectar una aceleración en una dirección de eje Z. En el documento D1, estos sensor y conmutador detectan al menos una de
15 una cantidad (por ejemplo, cantidad de inclinación, cantidad de movimiento, cantidad de impacto o similares) y una dirección (por ejemplo, dirección de inclinación, dirección de movimiento, dirección de impacto o similares) de un cambio aplicado a la carcasa. De acuerdo con el documento D1, un programa de simulación proporciona una simulación de tal modo que un estado de un espacio de juego se cambia en relación con al menos una de la
20 cantidad y la dirección del cambio aplicado a la carcasa.

- Cada vez más dispositivos electrónicos tales como teléfonos móviles, reproductores de MP3, Asistentes Digitales Personales (PDA) se están volviendo cada vez más pequeños al tiempo que tienen cada vez más información almacenada y/o accesible a través de los mismos. Los usuarios confían en estos dispositivos y se están volviendo
25 cada vez más dependientes de los mismos. Debido a la complejidad de los dispositivos, a veces puede resultarle difícil a un usuario aprender y entender los mismos. Hay una plétora de funcionalidades, características, aplicaciones y accesos directos disponibles a través de una amplia diversidad de entradas de usuario tales como entrada de toque, entrada de teclado numérico, entrada de lápiz y, recientemente, también entrada basada en sensores tal como entrada de golpecitos o entrada de movimientos de inclinación usando acelerómetros. Por lo tanto, la
30 tremenda cantidad de posibilidades se puede volver desconcertante para un usuario y aumentar el esfuerzo requerido para aprender y entender el dispositivo de forma significativa.

- Los entornos en donde estos dispositivos se usan también se están volviendo cada vez más frenéticos con demandas más altas sobre un acceso rápido y multitarea, y no solo en un entorno profesional sino también en entornos privados y sociales. Por ejemplo, un usuario debería ser capaz de leer un libro, al tiempo que sostiene una
35 bolsa de la compra en un vagón de metro traqueteante y seguir pudiendo acceder a la información almacenada en el dispositivo para que el dispositivo pudiera cumplir con los muchos requisitos planteados sobre el mismo por un usuario que compra y usa el dispositivo. Esto requiere que el dispositivo sea fácil de aprender e intuitivo de usar.

40 **Divulgación de la invención**

De acuerdo con varias pero no necesariamente todas las realizaciones de la invención, se proporciona un dispositivo, un método y un medio legible por ordenador de acuerdo con las reivindicaciones adjuntas.

- 45 En este contexto, un objetivo de la presente solicitud es la provisión de un dispositivo y un método que supera o al menos reduce los inconvenientes indicados anteriormente al proporcionar una interfaz de usuario, un dispositivo y un método que es sencillo de aprender y entender e intuitivo de usar.

- Este objetivo se logra al proporcionar un módulo de interfaz de usuario que comprende un visualizador y un detector de movimiento dispuesto para detectar un movimiento, estando dispuesto dicho módulo para visualizar un movimiento emulado de una representación gráfica de un primer objeto en dicho visualizador de acuerdo con un movimiento detectado por dicho detector de movimiento.
50

- Mediante el control de una interfaz de usuario con movimientos a través de cualquier objeto, se logra una interfaz de usuario que es fácil de usar.
55

En una realización, el primer objeto comprende una característica física y dicho módulo de interfaz de usuario comprende adicionalmente un módulo de control dispuesto para generar dicho movimiento emulado sobre la base de dicha característica física.
60

Al tener un modelo físico y basando los movimientos en este modelo y, por lo tanto, en unas características físicas de los objetos, el comportamiento del objeto se vuelve más intuitivo y sencillo de entender por un usuario, volviendo de ese modo la interfaz de usuario sencilla de aprender y usar.

- 65 En una realización, la característica física es una masa para emular la inercia de dicho objeto, en una realización, la característica física es un vector velocidad y, en una realización, la característica física es una extensión. Esto

posibilita que se modele un comportamiento dinámico realista según leyes físicas.

5 En una realización, el visualizador tiene un área de visualización y la representación gráfica de dicho objeto es más pequeña que el área de visualización. Esto posibilita una visión de conjunto de más de un objeto y una indicación visual del movimiento del objeto y su interacción con otros objetos visualizados en el mismo visualizador y el comportamiento emulado resultante. Esto aumenta adicionalmente la asociación intuitiva entre un movimiento físico y una acción resultante.

10 En una realización, las características físicas comprenden un coeficiente elástico y un coeficiente de amortiguación y en donde dicha generación de dicho movimiento emulado se basa en una dinámica de resorte amortiguado.

15 En una realización, la generación de dicho movimiento emulado comprende determinar una posición actualizada a partir de una posición original dependiendo de dichas características físicas y dicho movimiento detectado por el detector de movimiento y en donde dicha visualización de dicho movimiento emulado comprende visualizar dicho objeto en dicha posición actualizada.

20 En una realización, la interfaz de usuario se dispone adicionalmente para visualizar representaciones gráficas de al menos un objeto adicional, en donde dicho al menos un objeto adicional tiene unas características físicas y dicha generación de dicho movimiento emulado de dicho primer objeto se basa adicionalmente en dichas características físicas de dicho al menos un objeto adicional. En una interfaz de usuario con más de un objeto, se puede establecer una dependencia o relación conjunta entre entidades de datos, haciendo de este modo la interfaz de usuario más versátil.

25 En una realización, el primer objeto y/o dicho al menos un objeto adicional tiene un valor de datos en donde dicho módulo de interfaz de usuario se dispone adicionalmente para visualizar dicho valor de datos como parte de dicha representación gráfica de dicho objeto. Esto hace sencillo visualizar información a un usuario.

30 En una realización, la interfaz de usuario se dispone adicionalmente para ejecutar una función asociada a un patrón de movimiento y en donde dicho módulo de control se dispone para ejecutar dicha función sobre dicho objeto tras la detección de dicho patrón de movimiento. Esto hace que el control y la activación de operaciones o funciones sea fácil de iniciar debido a que no se requiere movimiento preciso alguno, como pulsar una tecla específica, y la interfaz de usuario es más intuitiva debido a que la función se conecta visiblemente a un objeto.

35 En una realización, el patrón de movimiento se corresponde con un movimiento de lanzamiento y dicha función de movimiento es una operación de suprimir y, en una realización, el patrón de movimiento se corresponde con un movimiento hacia arriba y dicha función de movimiento es una operación de abrir.

40 En una realización, un objeto tiene una función asociada. Asociar directamente una función a un objeto hace que sea más intuitivo ejecutar la función y no atesta el área de visualización con etiquetas funcionales.

45 En una realización, un patrón de movimiento se corresponde con una colisión emulada entre un objeto que tiene un valor de datos y un objeto que tiene una función de objeto asociada, y en donde dicha interfaz de usuario se dispone para ejecutar dicha función sobre dicho valor de datos tras la detección de dicha colisión emulada. Dar lugar a una colisión o forzar que unos objetos se toquen proporciona una imagen mental de cómo ejecutar una función que correlaciona la función y el valor de datos de una forma fácil y sencilla de aprender y recordar.

50 En una realización, la función es cualquiera tomada de entre el grupo que comprende: una operación de suprimir, una operación de llamar, una operación de enviar mensaje, una operación de posponer, una operación de abrir y una operación de leer en voz alta.

En una realización, el objeto es un objeto de área. En una realización, el objeto adicional es estático.

55 En una realización, el visualizador y el detector de movimiento se disponen en una y la misma carcasa que aumenta adicionalmente la conexión o acoplamiento intuitivo entre los movimientos y los movimientos resultantes modelados de forma realista y las acciones asociadas.

Los objetivos anteriores también se logran al proporcionar un dispositivo que tiene un módulo tal como anteriormente. En una realización, el dispositivo es un terminal móvil.

60 Los objetivos anteriores también se logran al proporcionar un método para controlar una interfaz de usuario que comprende un visualizador, un detector de movimiento y al menos un objeto, comprendiendo dicho método detectar un movimiento, emular un movimiento para dicho al menos un objeto de acuerdo con dicho movimiento, en donde se determina una posición actualizada, y visualizar dicho al menos un objeto en dicha posición actualizada, en donde la emulación de movimiento se basa en una característica física de dicho objeto. Esto prevé un método dinámico para controlar una interfaz de usuario que es sencilla de controlar e intuitiva de aprender al tiempo que versátil y ampliable.

65

En una realización, el método comprende adicionalmente determinar si una colisión entre un primer y un segundo objeto es causada por dicho movimiento emulado y emular dicho movimiento en consecuencia. El concepto de colisión simula o modela dependencias conjuntas entre objetos de una forma que es sencilla de controlar e intuitiva de usar, aprender y entender.

5 En una realización, un objeto tiene una función asociada que se corresponde con un patrón de movimiento y dicho método comprende adicionalmente determinar si dicho movimiento detectado completa dicho patrón de movimiento y, si es así, ejecutar dicha función. Esto asocia un objeto a una funcionalidad de una forma realista e intuitiva.

10 En una realización, el patrón de movimiento es una colisión.

En una realización, el al menos un objeto tiene un valor de datos y dicho método comprende adicionalmente ejecutar dicha función asociada sobre dicho valor de datos tras un patrón de movimiento completado. Esto proporciona una forma en la que unos datos están relacionados con una función de una forma muy intuitiva.

15 En una realización, la ejecución de dicha función asociada comprende generar un objeto adicional.

De acuerdo con otro aspecto de la presente solicitud, los objetivos anteriores se logran al proporcionar un módulo de interfaz de usuario que comprende unos medios de detección de movimiento para detectar un movimiento, y unos medios de visualización para visualizar un movimiento emulado de una representación gráfica de un primer objeto de acuerdo con un movimiento detectado por dichos medios de detección de movimiento.

20 En una realización, el primer objeto comprende una característica física y dicho módulo de interfaz de usuario comprende adicionalmente unos medios de controlador para generar dicho movimiento emulado sobre la base de dicha característica física.

En una realización, el primer objeto y/o al menos un objeto adicional tiene un valor de datos en donde dicho módulo de interfaz de usuario comprende adicionalmente unos medios de visualización para visualizar dicho valor de datos como parte de dicha representación gráfica.

30 En una realización, la interfaz de usuario comprende adicionalmente unos medios para ejecutar una función asociada a un patrón de movimiento sobre dicho objeto tras la detección de dicho patrón de movimiento.

35 Una interfaz de usuario tal como esta tiene los mismos beneficios que se han explicado para la interfaz de usuario y el método anteriores.

De acuerdo con otro aspecto de la presente solicitud, los objetivos anteriores se logran al proporcionar un medio legible por ordenador que incluye al menos código de programa informático para controlar una interfaz de usuario que comprende un visualizador, un detector de movimiento y al menos un objeto, comprendiendo dicho medio legible por ordenador código de software para recibir un movimiento a través de dicho detector de movimiento, código de software para emular un movimiento para dicho al menos un objeto, y código de software para visualizar dicho movimiento de dicho al menos un objeto.

45 En una realización, el medio legible por ordenador comprende adicionalmente código de software para determinar si dicho movimiento completa un patrón de movimiento y, si es así, para ejecutar una función asociada que se corresponde con dicho patrón de movimiento.

50 Esto posibilita que un dispositivo se beneficie de las ventajas que se han descrito anteriormente y también más adelante en la descripción detallada al ejecutar el código de software.

En un aspecto, los objetivos se logran mediante un dispositivo que incorpora y que pone en práctica un medio legible por ordenador de acuerdo con lo anterior.

55 Objetos, características, ventajas y propiedades adicionales del dispositivo, el método y el medio legible por ordenador de acuerdo con la presente solicitud se harán evidentes a partir de la descripción detallada.

Breve descripción de los dibujos

60 En la siguiente porción detallada de la presente descripción, las enseñanzas de la presente solicitud se explicarán con más detalle con referencia a las realizaciones a modo de ejemplo mostradas en los dibujos, en los que:

la figura 1 es una visión de conjunto de un sistema de telecomunicaciones en el que un dispositivo de acuerdo con la presente solicitud se usa de acuerdo con una realización,
la figura 2 es una vista frontal plana de un dispositivo de acuerdo con una realización,
65 la figura 3 es un diagrama de bloques que ilustra la arquitectura general de un dispositivo de la figura 1 de acuerdo con la presente solicitud,

la figura 4 a, b, c, d, e, f, g, h e i son vistas frontales planas de un dispositivo de acuerdo con una realización,
 la figura 5 es una vista esquemática de un modelo usado en una realización,
 la figura 6 a y b son diagramas de flujo que describen un método, cada uno de acuerdo con una realización,
 la figura 7 es una vista esquemática de un modelo usado en una realización, y
 5 la figura 8 es una vista frontal plana de un dispositivo de acuerdo con una realización.

Descripción detallada de realizaciones preferidas

10 En la siguiente descripción detallada, el dispositivo, el método y el producto de software de acuerdo con las enseñanzas para la presente solicitud en forma de teléfono celular/móvil serán descritos por las realizaciones. Se debería hacer notar que, aunque solo se describe un teléfono móvil, las enseñanzas de la presente solicitud también se pueden usar en cualquier dispositivo electrónico tal como en dispositivos electrónicos portátiles tales como portátiles, PDA, terminales de comunicaciones móviles, libros electrónicos y blocs de notas y otros dispositivos electrónicos que ofrecen acceso a información.

15 La figura 1 ilustra un ejemplo de un sistema de telecomunicaciones celular en donde se pueden aplicar las enseñanzas de la presente solicitud. En el sistema de telecomunicaciones de la figura 1, diversos servicios de telecomunicaciones tales como llamadas de voz celulares, navegación www/wap, videollamadas celulares, llamadas de datos, transmisiones de fax, transmisiones de música, transmisiones de imágenes fijas, transmisiones de vídeo,
 20 transmisiones de mensajes electrónicos y comercio electrónico se pueden realizar entre un terminal móvil 100 de acuerdo con las enseñanzas de la presente solicitud y otros dispositivos, tales como otro terminal móvil 106 o un teléfono estacionario 132. Se ha de hacer notar que, para diferentes realizaciones del terminal móvil 100 y en diferentes situaciones, se pueden encontrar disponibles, o no, unos diferentes de los servicios de telecomunicaciones a los que se ha hecho referencia anteriormente; las enseñanzas de la presente solicitud no se limitan a conjunto particular alguno de servicios a este respecto.

25 Los terminales móviles 100, 106 se conectan a una red de telecomunicaciones móviles 110 a través de enlaces de RF 102, 108 por medio de las estaciones base 104, 109. La red de telecomunicaciones móviles 110 puede ser en cumplimiento de cualquier norma comercialmente disponible de telecomunicaciones móviles, tal como GSM, UMTS, D-AMPS, CDMA2000, FOMA y TD-SCDMA.

30 La red de telecomunicaciones móviles 110 se conecta operativamente a una red de área extensa 120, que puede ser Internet o una parte de la misma. Un servidor de Internet 122 tiene un almacenamiento de datos 124 y se conecta a la red de área extensa 120, como lo es un ordenador de cliente de Internet 126. El servidor 122 puede alojar un servidor www/wap capaz de servir contenido www/wap del terminal móvil 100.

35 Una red telefónica pública conmutada (PSTN) 130 se conecta a la red de telecomunicaciones móviles 110 de una manera familiar. Diversos terminales de teléfono, incluyendo el teléfono estacionario 132, se conectan a la PSTN 130.

40 El terminal móvil 100 también es capaz de comunicarse localmente por medio de un enlace local 101 con uno o más dispositivos locales 103. El enlace local puede ser cualquier tipo de enlace con un alcance limitado, tal como Bluetooth, un enlace de Bus Serie Universal (USB), un enlace de Bus Serie Universal Inalámbrico (WUSB), un enlace de red de área local inalámbrica de IEEE 802.11, un enlace serie de RS-232, etc. Los dispositivos locales 103
 45 por ejemplo pueden ser diversos sensores que pueden comunicar valores de medición del terminal móvil 100 a través del enlace local 101.

50 Una realización 200 del terminal móvil 100 se ilustra con más detalle en la figura 2. El terminal móvil 200 comprende un altavoz o auricular 202, un micrófono 205, un visualizador principal o primer visualizador 203 y un conjunto de teclas 204 que puede incluir un teclado numérico 204a de tipo ITU-T común (un teclado alfanumérico que representa los caracteres "0"- "9", "*" y "#") y ciertas otras teclas tales como las teclas programables 204b, 204c y una palanca de mando 211 u otro tipo de dispositivo de entrada de navegación. El teléfono móvil también puede comprender un visualizador adicional, un visualizador secundario dispuesto en la cara posterior del teléfono móvil 200 (mostrado en la figura 4).

55 En lo sucesivo, se hará referencia a un visualizador como que se encuentra inactivo cuando este o la imagen de visualización en el mismo no sea visible para un usuario y este no se use para visualizar ninguna información u otros datos. Un experto observará que el visualizador puede aún estar funcionando aunque usando menos potencia que cuando el mismo se encuentra activo, es decir, la imagen visualizada es visible para un usuario.

60 La estructura interna de componentes, software y protocolo del terminal móvil 200 se describirá a continuación con referencia a la figura 3. El terminal móvil tiene un controlador 300 que es responsable del funcionamiento global del terminal móvil y se puede poner en práctica mediante cualquier CPU ("Unidad de Procesamiento Central"), DSP ("Procesador Digital de Señales") o cualquier otro dispositivo electrónico de lógica programable comercialmente disponible. El controlador 300 tiene asociada una memoria electrónica 302 tal como memoria RAM, memoria ROM,
 65 memoria EEPROM, memoria flash, o cualquier combinación de las mismas. La memoria 302 es usada para diversos fines por el controlador 300, siendo uno de los mismos almacenar datos usados por e instrucciones de programa

para diversos software en el terminal móvil. El software incluye un sistema operativo en tiempo real 320, unas unidades de accionamiento para una interfaz hombre - máquina (MMI) 334, un manejador de aplicaciones 332 así como diversas aplicaciones. Las aplicaciones pueden incluir una aplicación de calendario 350, una aplicación de bloc de notas 360, así como diversas otras aplicaciones 370, tales como aplicaciones para llamadas de voz, videollamadas, envío y recepción de mensajes de servicio de mensajes cortos (SMS), mensajes de servicio de mensajes multimedia (MMS) o correo electrónico, navegación web, una aplicación de mensajería instantánea, una aplicación de guía telefónica, una aplicación de calendario, una aplicación de panel de control, una aplicación de camera, uno o más videojuegos, una aplicación de bloc de notas, etc.

10 La MMI 334 también incluye uno o más controladores de hardware que, junto con las unidades de accionamiento de MMI, cooperan con el primer visualizador 336/203, el visualizador secundario 340, el teclado numérico 338/204 y un sensor o detector de movimiento 342 así como diversos otros I/O dispositivos tales como micrófono, altavoz, vibrador, generador de tonos de llamada, indicador de LED, etc. Tal como es conocido por un experto en la materia, el detector de movimiento 342 se puede poner en práctica con, por ejemplo, un sensor de aceleración u otro sensor capaz de detectar un movimiento. Ejemplos de tales sensores son los sistemas micro-electromecánicos (MEMS), sistemas nano-electromecánicos (NANS), sensores de aceleración piezoeléctricos, sensores de vibración, acelerómetros de láser, acelerómetros de resonancia, acelerómetros de galga extensométrica, acelerómetros capacitivos basados en masa y resorte, acelerómetros electromecánicos, acelerómetros ópticos o giróscopos. Tal como es comúnmente conocido, el usuario puede operar el terminal móvil a través de la interfaz hombre - máquina así formada.

25 El software también incluye diversos módulos, pilas de protocolos, unidades de accionamiento, etc., que se designan comúnmente como 330 y que proporcionan servicios de comunicación (tales como transporte, red y conectividad) para una interfaz de RF 306 y, opcionalmente, una interfaz de Bluetooth 308 y/o una interfaz de IrDA 310 para conectividad local. La interfaz de RF 306 comprende una antena interna o externa así como una circuitería de radio apropiada para establecer y mantener un enlace inalámbrico con una estación base (por ejemplo, el enlace 102 y la estación base 104 en la figura 1). Tal como es bien conocido por un experto en la materia, la circuitería de radio comprende una serie de componentes electrónicos analógicos y digitales, formando conjuntamente un receptor y transmisor de radio. Estos componentes incluyen, filtros de paso de banda, amplificadores, mezcladores, osciladores locales, filtros de paso de bajos, convertidores AD/DA, etc.

El terminal móvil también tiene una tarjeta de SIM 304 y un lector asociado. Tal como es comúnmente conocido, la tarjeta de SIM 304 comprende un procesador así como memoria de datos y de trabajo local.

35 La figura 4 muestra un dispositivo en forma de teléfono móvil 400 de acuerdo con la presente solicitud. El teléfono móvil 400 tiene una interfaz de usuario que posibilita que un usuario interactúe con el dispositivo a través de un visualizador 402 y una cierta entrada. Esta entrada puede ser o bien un teclado numérico 204, como en la figura 2, o bien un sensor de movimiento 404 tal como un acelerómetro. El teclado numérico puede tener teclas físicas, teclas programables, teclas virtuales o cualquier combinación de estas dependiendo de las elecciones de diseño. También es plausible y sumamente beneficiosa una combinación de sensores de movimiento y teclas. En la siguiente descripción, solo se hará énfasis en el detector de movimiento 402, pero se debería entender que las enseñanzas en el presente documento funcionan igualmente bien con cualquier combinación de un detector de movimiento y teclas tal como anteriormente.

45 Un detector de movimiento 404 (342 en la figura 3) se dispone en el teléfono móvil 400. El detector de movimiento se muestra en líneas de trazo discontinuo debido a que el mismo es parte de la circuitería interna del teléfono móvil y no es visible tal como se ha divulgado anteriormente. Tras la detección de un movimiento o un patrón de movimiento, una señal de detección es generada y procesada por el controlador (300 en la figura 3) del teléfono móvil 400. Se debería entender que el detector de movimiento 404 podría comprender tanto un sensor para detectar el movimiento físico como un controlador para diferenciar entre diferentes movimientos físicos y patrones de movimiento, controlador que podría ser el mismo que el controlador 300 del teléfono móvil 400.

50 Se debería hacer notar que, debido a que el visualizador 402 y el detector de movimiento 404 se disponen en la misma carcasa de un dispositivo, en la presente realización un teléfono móvil 400, la interfaz de usuario puede responder a las fuerzas físicas a las que se somete el dispositivo y esta puede servir como parte de una realimentación multimodal.

60 Tal como es comúnmente conocido, el visualizador 402 se usa para retransmitir información a un usuario y para posibilitar que el mismo interactúe con el teléfono móvil 400. La información se retransmite al visualizarse en el visualizador 402. Tradicionalmente, esta información se visualiza de forma estática. Debido a que la propia naturaleza del control de movimiento de un teléfono móvil es dinámica, puede ser confuso para un usuario relacionar los movimientos de control dinámico con las funciones estáticas y el comportamiento del teléfono móvil 400.

65 Para superar esto, un modelo dinámico, con elementos u objetos de interfaz de usuario, se usa para crear una relación dinámica entre objetos que se usan para retransmitir información. Los objetos 406, 407 y 408 pueden tener diferentes rasgos o características que se analizarán con más detalles posteriormente. Tres ejemplos de

características son un valor de datos, una función asociada y si el objeto es, o no, móvil. Un objeto con un valor de datos se usa habitualmente para visualizar este el valor de datos. Se usa un objeto con una función asociada para informar al usuario de que la función se encuentra disponible y, si se efectúa, para iniciar o ejecutar la función. Los objetos no móviles se pueden usar para interactuar con un objeto móvil y para imponer algunas restricciones de movimiento sobre la representación gráfica de la interfaz de usuario.

Para aumentar adicionalmente la comprensión de la conexión entre el movimiento físico y el movimiento emulado resultante de un objeto, se necesita un comportamiento físicamente realista. Si el movimiento emulado resultante no es realista, este será confuso y desconcertante para un usuario.

El modelo dinámico se basa en que todos los objetos tienen algunas características físicas y que estos se ven afectados por los movimientos detectados a través de un sensor de movimiento de la misma forma en la que se vería afectado un objeto físico normal que tuviera las mismas características.

Las características físicas pueden ser una masa y una extensión física. La masa emulará la inercia en el objeto, determinando cómo de rápido acelera y rota el mismo así como el cómo se comporta éste tras una colisión con otros objetos. La extensión física define las fronteras del objeto y se usa para determinar si un objeto colisiona, o no, con otro.

Algunos objetos carecen de masa y no tienen frontera alguna, esto los hace puramente gráficos y los mismos pueden acelerar y moverse a cualquier velocidad y no colisionan, meramente se solapan. Tales objetos se pueden usar para modelar entidades de información para las cuales no se ha de emprender función alguna tal como etiquetas o notificaciones de tiempo. Las etiquetas para teclas programables no deberían ser móviles excepto tal como se explica con referencia a la figura 7a debido a que las mismas se deberían mantener cerca de la tecla programable asociada.

Otras características posibles son el rozamiento, la elasticidad (elasticidad superficial y elasticidad volumétrica).

Para modelar de forma realista todos los aspectos físicos de los objetos en movimiento y su interacción entre sí, el modelo dinámico hace uso de: modelado de la física, modelado de rozamiento, modelado de colisión de múltiples objetos, realimentación basada en la detección de colisiones, modelado de centro de masa y otras técnicas usadas para modelar un movimiento dinámico. En una realización, se usaron algoritmos para modelar una dinámica de cuerpo rígido, tales como momento lineal, momento angular y par de fuerzas, unos métodos de simulación para movimientos usando un modelo basado en velocidad de multiplicadores de Lagrange y un modelo de contacto y de rozamiento sobre la base del solucionador de LCP de Dantzig.

Tal como es conocido por un experto en la materia, los problemas de optimización se pueden investigar usando el método de los multiplicadores de Lagrange, que es un método para hallar los extremos de una función de varias variables sometida a una o más restricciones; este es la herramienta básica en la optimización restringida no lineal. Este reduce el hallazgo de puntos estacionarios de una función restringida en n variables con k restricciones al hallazgo de puntos estacionarios de una función no restringida en $n + k$ variables.

Tal como también es bien conocido por un experto en la materia, el solucionador de LCP de Dantzig soluciona los problemas de modelado de rozamiento mediante la aplicación de Programación Complementaria Lineal, LCP, según fue presentada por George Dantzig.

Este modelo dinámico requiere mucha potencia de cálculo y se usa una optimización para dispositivos con recursos bajos tales como un teléfono móvil. La optimización se basa en sustituir presentaciones de coma flotante con presentaciones de coma fija en cálculos que van a ser hechos por el controlador 300. Esto se puede realizar al realizar la sustitución de coma flotante a coma fija directamente como en la tabla 1 en donde Q denota la mantisa de la operación de coma fija y " $>>$ " la entidad que se va a sustituir. Para cálculos rápidos, se usan valores de 16 o 32 para Q . También son posibles otros valores, tal como será evidente a un experto dependiendo del controlador usado para la sustitución. Una sustitución directa ahorra tiempo, debido a que no es necesaria llamada de procedimiento alguna.

La figura 5 muestra un modelo dinámico 500 para una interfaz de usuario, UI, que tiene un controlador 510. El controlador 510 se dispone para crear y controlar uno o más objetos 540. Estos objetos 540 tienen algunas características que son comunes a todos los objetos y estas son su posición y su extensión. Estas características son necesarias para generar una representación gráfica de un objeto. Algunos objetos no tienen representación alguna sino la posición y la extensión del objeto sigue siendo necesaria para determinar su interacción con otros objetos. Tales objetos se denominan en lo sucesivo en el presente documento objetos de área. Otras características se muestran en la tabla 2 junto con sus fines. El modelo 500 también tiene un detector de movimiento 520 desde el cual se proporciona información del controlador 300 con respecto a los a los movimientos que está sometido el dispositivo y las fuerzas que se aplican al mismo. El detector de movimiento 520 se puede poner en práctica a través de un sensor de aceleración. Se puede usar un integrador para integrar la aceleración detectada, y a partir de esta, generar un movimiento. Estos movimientos pueden ser un patrón de movimiento a través del cual un usuario

controla uno o más objetos. El modelado del movimiento de los objetos se ha analizado y explicado anteriormente.

En una realización, también se usa un detector de colisiones 530 en el modelo 500. El detector de colisiones 520 se podría hacer parte del controlador 510, pero en la presente descripción se lo describe como una entidad separada para ilustrar mejor su funcionalidad. Si dos objetos se encuentran activos, por ejemplo uno estacionario y uno dinámico, y se detecta un movimiento, el detector de colisiones empezará a supervisar si se solapa cualquiera de las posiciones y la extensión de los objetos. Una fórmula simplificada para una colisión es:

$$\text{pos A} + \text{EXTENSIÓN A} \subseteq \text{pos B} + \text{EXTENSIÓN B} \text{ O}$$

$$\text{pos B} + \text{EXTENSIÓN B} \subseteq \text{pos A} + \text{EXTENSIÓN A} \Rightarrow \text{COLISIÓN}$$

en donde pos es una posición que se añade uniformemente a la extensión de un objeto.

La fórmula se ha de entender como si cualquier parte del objeto A se solapa con cualquier parte del objeto B entonces hay una colisión.

Los objetos pueden ser o bien estacionarios, que se pueden modelar como si tuvieran una masa infinita, o bien dinámicos, que tienen una masa finita o bien puramente gráficos, que tienen una masa de 0. Un objeto puramente gráfico será capaz de desplazarse a cualquier velocidad y solaparse con otros objetos y se usan para fines informativos tales como etiquetado.

Para los objetos dinámicos, un movimiento emulado se determina entre actualizaciones de un visualizador. El movimiento emulado es una función de la posición actual, inercia, masa o velocidad actual, la orientación del cuerpo, el rozamiento de la superficie del objeto o la superficie de lo circundante y la fuerza y a partir de ello derivarse la aceleración aplicada a la que se somete el objeto. Una posición actualizada se calcula a partir de la posición actual usando los valores característicos conocidos y las técnicas de modelado descritas en el presente documento. La posición, la velocidad y la velocidad angular se pueden expresar mediante vectores y la orientación se puede expresar como una matriz de rotación 3x3.

Características tales como la masa también se pueden expresar como una matriz. Para la masa, se puede usar una constante para describir la masa total. Un vector de centro para expresar el centro de masa visto desde un punto de referencia y una matriz de inercia que describe como se distribuye la masa.

Un objeto puede tener un valor de datos asignado al mismo. Estos valores de datos se pueden usar para notificar del usuario acerca de determinados eventos o los mismos pueden tener otro significado.

Un objeto 540 puede tener, adicionalmente o como alternativa, una o más funciones asociadas al mismo. Cuando se activa un objeto, se ejecuta la función correspondiente. Ejemplos de tales funciones son: ABRIR, SUPRIMIR, CERRAR o POSPONER. Una función también se puede diseñar para funcionar sobre uno o más valores de datos, que o bien son portados por el propio objeto o por otro objeto. Si tiene lugar una colisión entre un objeto de valor de datos y un objeto de función, la función se ejecuta usando el valor de datos como un parámetro. Tal como se puede ver en la figura 5b, cuando colisionan un objeto de función 540A y un objeto de datos 540B, como resultado la función se ejecuta sobre los datos.

Si un objeto porta más de una función, la función que se va a ejecutar puede depender del objeto con el que se colisiona y también de los datos portados por el objeto con el que se colisiona.

Para los objetos especiales a los que se hace referencia como objetos de área explicados anteriormente, una colisión tiene lugar cuando otro objeto entra en el área, o dicho de otra forma, la extensión de las áreas se solapa con la extensión de otro objeto. Los objetos de área se pueden usar como objetos que portan funciones. Un ejemplo de un objeto de ese tipo es una zona de suprimir modelada a través de un objeto de área que tiene la función SUPRIMIR asociada.

La posición de un objeto podría ser tal que, cuando la representación gráfica se muestra en un visualizador, el objeto puede haber desaparecido parcial o completamente del visualizador, es decir, este no es visible debido a que el mismo se encuentra fuera del área visualizada.

Se debería entender que, en este modelo, una colisión se puede considerar como un patrón de movimiento y una función se va a ejecutar cuando se detecta un determinado patrón de movimiento. El patrón de movimiento puede ser entonces un movimiento controlado o una colisión. Por lo tanto, se supone que las colisiones son controladas por un usuario para iniciar una determinada acción. También se debería entender que un movimiento puede ser parte de un patrón de movimiento (posiblemente la única parte del patrón de movimiento) y el patrón de movimiento es completado por dicho movimiento que da lugar a que se emprenda o se ejecute cualquier acción o función correspondiente.

Por ejemplo, al tener un área de objeto virtualmente situada en la parte de arriba de la pantalla y asignada a la misma la función SUPRIMIR, otro objeto se puede suprimir fácilmente al lanzar este hacia arriba o poner el teléfono boca abajo y permitir que el objeto caiga fuera de la interfaz de usuario y, por lo tanto, se suprima. Para proteger frente a una supresión accidental, se puede asignar del objeto de área un rozamiento que ralentizaría cualquier objeto que pasara por encima de la misma y, por lo tanto, evitaría la ejecución no deseada de la operación asignada, supresión en este caso.

Una función también se podría activar si el objeto portador se somete a un patrón de movimiento, es decir, una secuencia de o un único movimiento controlado que es de naturaleza específica. Los ejemplos pueden ser desechar (expresado mediante una aceleración fuerte a lo largo de un periodo de tiempo), elevación (expresado mediante una aceleración fuerte hacia arriba seguida por una aceleración hacia abajo) o un golpecito (expresado mediante un choque o pulso corto).

Para proteger frente a una ejecución accidental de funciones resultantes de colisiones y patrones de movimiento aleatorios, se puede crear un objeto de confirmación y visualizarse, induciendo del usuario a confirmar si la acción es, o no, deliberada. Un golpecito del dispositivo, induciendo de ese modo un choque ligero que es detectado por el detector de movimiento ejecutará la función. Como alternativa, se puede usar un doble golpecito para aceptar la acción y un único golpecito para rechazar o cancelar la misma.

En una realización, una realimentación táctil es generada por el detector de colisiones tan pronto como se detecta una colisión. Una realimentación táctil de ese tipo se puede encontrar en forma de vibración. Para algunos fines, también es posible dar un choque eléctrico pequeño. Naturalmente, también son posibles realimentaciones visibles y audibles o bien solas o bien en cualquier combinación. Si colisionan dos objetos, se puede generar un tipo de vibración y, si un objeto colisiona con un objeto de pared, se puede generar otro tipo de vibración. Por lo tanto, un usuario puede diferenciar entre diferentes eventos que tienen lugar en la pantalla y ser capaz de controlar el dispositivo incluso sin mirar. Especialmente si la interfaz de usuario está equipada con un codificador de texto a habla u otro codificador de voz de tal modo que la información se puede leer en voz alta del usuario.

Para ilustrar con un ejemplo: un dispositivo recibe un mensaje entrante y se alerta al usuario con una señal audible. Debido a que el usuario está ocupado en parte con otra tarea, pero quiere aún así ver cuál es el mensaje, el usuario puede sacar rápidamente el teléfono, realizar una acción de apertura, que puede ser un movimiento de elevación que abrirá el objeto con la notificación del mensaje tal como se describirá posteriormente. El usuario conoce que, en el lado izquierdo del visualizador, hay un área para leer el mensaje en voz alta y ladea o inclina el teléfono hacia el lado izquierdo. Debido a otros factores que influyen en el movimiento (tales como si el usuario está montado en un tren) el objeto no acierta por poco el área de lectura en voz alta y una vibración seguida por un sonido estridente se emite a partir del dispositivo que informa al usuario de que el objeto está a punto de desplazarse a través de un área asociada a una operación de suprimir. El usuario reacciona a la realimentación y cambia el movimiento para que sea más claramente hacia el lado izquierdo y, cuando el objeto colisiona con el área de lectura de mensaje, otra vibración más suave y un sonido agradable se emite como una realimentación. Se informa al usuario de que la acción es correcta, o al menos de que la misma no tiene consecuencias irrevocables y, para finalizarla, exagera el movimiento con el fin de asegurar que se emprende la acción y que se ejecuta la función del área de pasar un texto a través de un codificador de texto a habla sobre el valor de datos del objeto, que es el texto del mensaje, y el mensaje se lee en voz alta mientras el objeto se encuentra en el área. Para silenciar la lectura en voz alta, el usuario simplemente mueve de nuevo el objeto fuera del área mediante el control del movimiento del dispositivo. Por lo tanto, un mensaje entrante se puede abrir, leer e incluso pausar mediante gestos manuales sencillos que son sencillos de aprender, intuitivos de usar y sin requerir mucha atención por un usuario.

En una realización, un dispositivo se somete a una función de bloqueo de tecla y, mientras está activa la función de bloqueo de tecla, solo se aceptará la notificación de acciones. Por ejemplo, si se activa un bloqueo de tecla y hay llamadas perdidas, solo se aceptarán las acciones para mostrar las llamadas perdidas. Cualquier acción para suprimir o devolver llamada será ignorada o cancelada por la función de bloqueo de tecla. Por lo tanto, si el movimiento que desencadena una acción es por casualidad, el estado del dispositivo será el mismo cuando el usuario dirige su atención a este que el que era antes de que se emprendieran las acciones.

Volviendo a la figura 4, se muestra una realización de una interfaz de usuario en un dispositivo 400 usando los modelos anteriores, haciendo la interfaz de usuario más intuitiva de usar con un sensor de movimiento 404. El dispositivo es, tal como se ha expuesto anteriormente, un teléfono móvil 400 y este tiene un visualizador 402. En la figura 4a, se visualizan tres objetos en el visualizador 402. Todos ellos son estacionarios, lo que se puede modelar al darles una masa infinita o simplemente al establecer un indicador que expone que estos son estacionarios. El primer objeto 406 se encuentra en forma de caja con un texto que indica o notifica a un usuario que hay 2 Llamadas perdidas. Si el usuario desea ver cuáles son las llamadas perdidas, pero en la actualidad no se encuentra disponible o no desea alcanzar algunas teclas de control (no mostradas), este simplemente hace un movimiento hacia arriba 420 que imita un patrón de movimiento para elevar la caja en el aire y de nuevo abajo. Esto desencadena que el detector de movimiento 404 envíe una señal que porta detalles que describen el movimiento, tal como vectores de velocidad, del controlador 300. El controlador determina que se ha iniciado una operación de abrir y ejecuta la misma. La operación de abrir crea otro cuarto objeto 409 y hace que el mismo flote hacia arriba desde el interior de

la caja. Este cuarto objeto 409 es un objeto dinámico y se está emulando su movimiento, con lo que la dirección depende del movimiento original detectado por el sensor y la velocidad depende de la masa del objeto 409 y la fuerza aplicada del teléfono móvil 400. Tal como se puede ver en la figura 4b, el objeto 409 tiene un valor de datos que se visualiza en el objeto 409 que indica que un contacto "John" había llamado a las 10:42. El usuario percibe esto pero elige no emprender acción alguna en este punto y decide suprimir la notificación. El usuario realiza un patrón de movimiento que simula un movimiento de lanzamiento 422 que es detectado por el detector de movimiento 404 y una señal se envía del controlador 300 que determina que se ha emprendido una acción de suprimir e inicia la acción de suprimir sobre el objeto 409. El movimiento emulado del objeto 409 se visualizará cuando este se esté arrojando hacia el lado del teléfono en la dirección en donde se hizo el movimiento de lanzamiento y el objeto 409 desaparece del visualizador 402 tal como se ve en la figura 4c. El patrón de movimiento de desechar el objeto causa, por lo tanto, una función para suprimir el objeto 409. Debido a que la representación gráfica del objeto 409 ha salido del visualizador 402, el visualizador es entonces como en la figura 4d con los tres objetos originales. El primer objeto 406 tiene entonces un valor de datos que notifica una llamada perdida y un movimiento de elevación repetida 420 hace que un quinto objeto 411 flote o vuele hacia arriba desde el objeto de caja de llamada perdida 406. Este quinto objeto también tiene un valor de datos que se visualiza que notifica que un contacto "Jane" realizó una llamada a las 9:34 tal como se muestra en la figura 4e. El usuario tiene entonces cuatro opciones. Este puede elegir suprimir, como hizo con la notificación previa, o puede guiar o controlar el movimiento del objeto 411 a través del detector de movimiento 404 mediante el control del movimiento del dispositivo 400 para colisionar con cualquiera de los objetos estacionarios visualizados. Los tres objetos estacionarios son el objeto de caja de llamada perdida 406, que indica ahora que no hay llamada perdida alguna, un objeto de llamada de teléfono 407 y un objeto de enviar mensaje 408. Los dos objetos, llamada de teléfono 407 y enviar mensaje 408, tienen funciones asociadas a los mismos, funciones que son evidentes a partir de la apariencia de la representación gráfica del objeto. Tras una colisión con un objeto de ese tipo, se iniciará la función asociada. El primer objeto 406 también tiene una función asociada que es almacenar cualquier objeto que colisione con el mismo. El usuario elige llamar del contacto mediante control o guiado, a través de un movimiento o patrón de movimiento 424, del quinto objeto 411 para colisionar con el objeto de llamada de teléfono 407. Cuando los dos objetos 407 y 411 colisionan en la figura 4f, la colisión se detecta en el controlador y la función asociada del objeto de llamada de teléfono 407 se ejecuta sobre el valor de datos asociado al quinto objeto 411. Debido a que el usuario ha elegido no guardar el objeto de notificación 411, se suprime el objeto de caja de llamada perdida 406.

Como alternativa, la operación de suprimir iniciada por el movimiento de lanzamiento 422 en la figura 4b y c se puede determinar como una colisión entre el objeto 409 y una pared lateral del visualizador 402 modelada como si no tuviera superficie alguna, por lo que no tiene lugar colisión alguna y en donde el movimiento emulado lleva claramente el objeto fuera del visualizador 402. La pared lateral se puede modelar como un objeto de área.

Como alternativa, si ambos objetos de notificación 409 y 411 se encontraban en el aire al mismo tiempo y ambos colisionaron con el símbolo de teléfono 407, se podría iniciar una llamada de tipo conferencia para ambos dos. Por lo tanto, una función también se puede establecer para operar sobre más de un valor de datos.

Otras notificaciones que se podrían visualizar son notificaciones de estado de llamadas entrantes, llamadas perdidas, mensajes recibidos (SMS, MMS, correos electrónicos, etc.), estado de descarga o un mensaje especial traído por una aplicación especialmente adaptada, por ejemplo dispuesta para mostrar dibujos animados de acuerdo con parámetros preestablecidos.

Como alternativa, se pueden asignar funciones a áreas del visualizador 402 tal como se puede ver en la figura 4g. Estas áreas se pueden modelar como objetos estacionarios que tienen una frontera no reflectante, es decir, capaz de solapamiento. En la figura 4g, se asignan dos operaciones a las paredes y estas operaciones son LLAMAR y SUPRIMIR. Las operaciones se pueden marcar con unas indicaciones 411 y 412. Tal como se ha descrito previamente, un usuario puede abrir la caja de llamada perdida 406 mediante un movimiento de elevación 420, tras lo cual la notificación de llamada 409 vuela hacia arriba, véase la figura 4h. Para suprimirlo, el usuario controla el objeto 409 a través de un patrón de movimiento 426 para colisionar con el área marcada SUPRIMIR 411 tal como se muestra en la figura 4i.

Ejemplos adicionales de funciones y los patrones de movimiento correspondientes se enumeran en la tabla 3.

En una realización, no es necesaria una colisión para iniciar una acción. Un objeto que tiene un valor de datos también puede tener una función asociada a un patrón de movimiento correspondiente. En el ejemplo anterior, el objeto 409 podría tener la función asociada de llamar del contacto a un patrón de movimiento correspondiente de un movimiento de agitación. Si un usuario deseara llamar del contacto visualizado, este simplemente agitaría el teléfono y el controlador ejecutaría la función asociada de llamar sobre el valor de datos que es el contacto.

Otro ejemplo es que, cuando un elemento descargado se ha descargado plenamente, aparece una caja de descarga. Un usuario puede abrir el elemento al elevarlo y, si el elemento es un archivo de medios, un movimiento de agitación ejecutaría un reproductor de medios para reproducir el archivo de medios descargado.

Lo más común es que, cuanto más avanzado sea el movimiento o patrón de movimiento, más seguro será suponer que el mismo se ha hecho a propósito, protegiendo de este modo frente a movimientos fortuitos. No obstante, un movimiento más avanzado o complicado es más difícil de lograr por un usuario. Qué patrones de movimiento y qué combinaciones se van a usar depende de aspectos de facilidad de uso para el teléfono si este es un modelo para trabajos pesados o un modelo de moda, etc., así como el grupo de usuarios previsto y una combinación de estos y otros factores.

La figura 6a muestra un método de acuerdo con las enseñanzas de la presente solicitud. En una primera etapa 610, se visualiza una notificación. Cuando se detecta un movimiento en un evento 615, se determina que el movimiento es un patrón de movimiento que se corresponde con una orden de ABRIR y se crea un objeto que porta los datos para la notificación. Su movimiento se emula en la etapa 620 y el objeto se visualiza en la etapa 630. Si se encuentran presentes otros objetos, sus movimientos también se emulan en la etapa 620 y el visualizador se actualiza con las posiciones nuevas en la etapa 630. A medida que se detectan movimientos adicionales en el evento 625, los movimientos se emulan y las posiciones se actualizan en la etapa 640. En la etapa 650, se determina si ha tenido lugar colisión alguna entre dos objetos. Si no hay colisión alguna, los objetos se visualizan en las posiciones actualizadas en la etapa 630. Si hay una colisión, la función asociada se determina en la etapa 660 y se ejecuta sobre los datos portados por el objeto en la etapa 670. Cuando se ejecuta la función, el método puede volver a la etapa 630 de visualizar los objetos tal como indica mediante la línea de trazo discontinuo. O, el método puede terminar dependiendo del número de objetos restantes y la naturaleza de la función.

La figura 6b muestra un método de acuerdo con las enseñanzas de la presente solicitud. En una primera etapa 601, un movimiento es detectado por un controlador 300 a través de un detector de movimiento 404. En la siguiente etapa 602, se emula un movimiento para un objeto 409 y se realiza una comprobación acerca de si un patrón de movimiento se ha completado en la etapa 603. Un patrón de movimiento puede ser, en este método, cualquiera de una colisión u otros patrones de movimiento. En la etapa 604, se comprueba si un objeto tiene una función asociada que se ejecutará entonces en la etapa 605. NO es necesario que un patrón de movimiento sea una colisión sino que, para un objeto que tiene tanto un valor de datos como una función, puede ser un patrón de movimiento específico que se corresponde con la función. Un ejemplo de esto se ha dado anteriormente con referencia a la figura 4 tal como cuando un usuario ha descargado un archivo de medios y lo reproduce al agitar el dispositivo.

La figura 7a muestra un modelo 700 usado para emular el comportamiento físico y el movimiento de un objeto en un eje. Para emular correctamente un movimiento tridimensional, se necesitaría un modelo para cada eje en la totalidad de los tres modelos requeridos. Este modelo se usa para objetos que tienen una coma fija en la interfaz de usuario, creando de este modo un comportamiento dinámico también para objetos estacionarios.

El modelo 700 se basa en un resorte amortiguado y un objeto 702 se conecta mediante un resorte 704 y un amortiguador 706 a una posición fija 708. De acuerdo con la ley de Hooke, el resorte tiene una constante elástica k y la fuerza F_s ejercida sobre el resorte es igual del vector de desplazamiento x multiplicado por esta constante elástica, $F_s = kx$. En la vida real, la constante elástica no es siempre constante sino que varía un poco con el desplazamiento y una mejor expresión sería, por lo tanto, el coeficiente elástico. El amortiguador 706 también se somete a una fuerza F_d que es igual a un coeficiente de amortiguación c multiplicado por una velocidad v de un objeto 702 acoplado con el amortiguador 706 en sentido negativo, $F_d = -cv$. Estas dos relaciones en combinación con la segunda ley del movimiento de Newton, que expone que la fuerza F aplicada a un objeto es igual a la masa m multiplicada por la aceleración a , $F = ma$, constituyen el modelo de resorte amortiguado que se usa para emular el movimiento de un objeto 702.

Usando este modelo, se puede hacer que un objeto estacionario se comporte de forma dinámica y que responda a órdenes inducidas por el usuario, y confirmando de ese modo las acciones emprendidas de una forma intuitiva. El modelo también se puede usar para anclar objetos estacionarios pero posibilitar que los mismos tengan un movimiento emulado como resultado de reacciones físicas, de modo que, cuando se detecta una colisión, el objeto estacionario podría reaccionar al agitar o botar como resultado del choque a partir de la colisión con el objeto en movimiento.

Este también se puede usar como una característica de protección a medida que los movimientos de algunos objetos para los cuales se aplica el resorte amortiguado se vuelven más restringidos y, para permitir que un objeto colisione con otro, se necesita un movimiento más enérgico y deliberado para superar la resistencia simulada ofrecida por el resorte. Dicho de otra forma, un efecto de rebote realista proporciona protección frente a movimientos erróneos debido a que se necesita un movimiento más deliberado para tocar o colisionar con otro objeto antes de rebotar.

El modelo de resorte amortiguado también se puede usar cuando se mueve un objeto. En la figura 7b se muestra un ejemplo ilustrativo. Un objeto 700 va a ser movido a una posición nueva 708 y la posición nueva 708 se establece para ser la coma fija 708 del modelo. Un resorte modelado se estira entonces automáticamente a un desplazamiento x igual a la distancia con respecto a la coma fija 708 y el objeto 700, sometiendo el objeto a una fuerza $F_s = kx$. Asimismo, también se estira un amortiguador modelado. Cuando se suelta el objeto 700, la fuerza F_x acelerará el objeto 700 hacia la coma fija 708 y, debido a esto, el objeto 700 empieza a moverse hacia la coma fija 708 y

desarrollar una velocidad v . El amortiguador ejercerá entonces una fuerza F_d que actúa en el sentido opuesto y de un tamaño igual a $F_d = cv$ que ralentizará el objeto 700. A medida que el objeto 700 se acerca a la coma fija 708, la fuerza aceleradora F_s disminuirá al mismo tiempo que crecerá la fuerza desaceleradora F_d . El objeto 700 sobrepasará la mayor parte de las veces la posición 708 de la coma fija y el resorte modelado forzará de nuevo el objeto 700 hacia la coma fija 708 y el objeto 700 descansará con el tiempo en la posición nueva 708. El tiempo para alcanzar la posición nueva por primera vez se denota como un tiempo de subida y el tiempo para que se asiente sobre la posición nueva se denota como un tiempo de asentamiento.

Este principio se puede aplicar a eventos de tecla y eventos de desplazamiento, control de gestos de movimiento y control de toques tal como entrada de golpecitos, por ejemplo.

También es posible mantener una fuerza constante virtual o emulada que afecte a todos los objetos dinámicos, de forma muy similar a una fuerza gravitatoria. La fuerza gravitatoria se puede dirigir o bien internamente en el teléfono, por ejemplo siempre hacia la parte de debajo de la pantalla o bien externamente como si coincidiera con la dirección del campo gravitatorio de la Tierra. Esto posibilitará que los objetos se comporten de una manera incluso más realista y también protejan frente a movimientos accidentales debido a que los objetos siempre pugnan por caer y un área, objeto o lugar seguro como el lugar de origen se debería situar en el punto más bajo de gravedad. Para que se iniciara una acción sería necesario, por lo tanto, que el movimiento de control superara la fuerza o el campo gravitatorio y, por lo tanto, se requeriría que fuera más deliberado. Por lo tanto, un objeto abierto que se deja tranquilo o inalterado simplemente caerá de vuelta a la caja de origen lo que, en la práctica, pondrá en práctica una característica de tiempo de expiración para cualquier acción emprendida por el usuario.

En una realización, un objeto de área que tiene una función SUPRIMIR asociada se dispone en la misma parte de arriba de un visualizador. La interfaz de usuario también se dispone con una fuerza gravitatoria dirigida externamente. Para suprimir un objeto, un usuario simplemente gira un dispositivo teniendo en mente una interfaz de usuario de ese tipo y permite que el objeto se suprima para verse afectado por la gravedad simulada y se hunda en el área de suprimir y se suprima. Esto conduce a la imagen mental de suprimir un objeto al verterlo fuera de la parte de arriba del dispositivo.

Se debería entender, para todas las realizaciones descritas en el presente documento, que un objeto también puede existir fuera de la representación gráfica, actuando por lo tanto la representación gráfica solo como una ventana a la interfaz de usuario.

También se debería hacer notar que, para algunos objetos, se determina que ha ocurrido una colisión solo si hay un solapamiento total de las extensiones de dos objetos. Esto se puede usar para áreas en donde el objeto que va a ser efectuado por una función de un área solo se ve afectado si este entra totalmente en el área. En el ejemplo anterior, esto querría decir que, si se dice que el área de suprimir se encuentra apenas fuera del área visualizada, el objeto tendría que salir de la pantalla completamente antes de suprimirse.

La figura 8 muestra un ejemplo de cómo afectaría una fuerza gravitatoria a la situación en la figura 4b. Un objeto se ha abierto desde el objeto de caja de llamada perdida 806 y está colgando actualmente en el aire. Una fuerza gravitatoria denotada F en la figura está actuando sobre el objeto de notificación 409, dando lugar a que este caiga de vuelta hacia la caja de llamada perdida 806. A menos que el usuario controle el movimiento de los objetos al mover el dispositivo 800 para hacer que este vuele hacia arriba hacia uno de los otros objetos 807, 808 para iniciar una función, el objeto de notificación 809 caerá de vuelta del objeto de caja de llamada perdida 806 y se guardará allí para un momento posterior. Por lo tanto, se logra un tiempo de expiración variable para la operación de apertura de la caja de llamada perdida, periodo de tiempo durante el cual es posible o bien realizar una acción sobre el objeto de notificación 809 producido o bien posponer o restablecer el tiempo de expiración al elevar adicionalmente el objeto 809.

También podría ser posible introducir un elemento, posiblemente como un objeto estacionario, en donde los otros objetos se estarán moviendo. Un elemento de ese tipo se puede usar para simular un entorno diferente que tenga propiedades particulares. Un ejemplo es simular un visualizador llena con agua en la que los objetos se desplazan flotando. Dependiendo de su extensión, su masa corporal y la densidad del agua o el elemento circundante, estos o bien se hundirán o bien flotarán, tal como se conoce a partir del principio de Arquímedes.

Por ejemplo, si una superficie o área de interfaz de usuario se diseña para llenarse a medias con agua, se logra un modelo para funciones solo aplicables a algunos objetos o en donde algunas funciones requieren una seguridad añadida. Una situación de ese tipo podría ser en donde todos los otros objetos flotan y la operación de suprimir se modela en un objeto en la parte de debajo. Para suprimir un objeto se requiere, por lo tanto, un movimiento forzado que empuje el objeto bajo el agua, y lo más probable es que no ocurra por casualidad.

Los elementos también se pueden usar para poner en práctica diferentes tiempos de expiración y funciones de tiempo de expiración para diferentes objetos al hacer que algunos floten y algunos se hundan y a unas velocidades que dependen de su densidad o masa. Por lo tanto, algunos objetos subirán hasta un objeto de función y algunos se hundirán a otro objeto de función si se dejan tranquilos o inalterados. La densidad determinará la velocidad, es decir,

el periodo de tiempo de expiración.

También es posible añadir objetos de textura que tengan un determinado rozamiento u otra característica que influya en los movimientos de los otros objetos así como que cree un efecto visual que se puede usar para aumentar la concepción visual de la representación gráfica de la interfaz de usuario y, por lo tanto, para fomentar la comprensión de cómo funciona la misma, conduciendo a una interfaz de usuario más sencilla y más intuitiva.

Debido a que el objeto se comporta como un objeto real, la conexión mental entre la operación y su efecto se vuelve más intuitivo y, por lo tanto, sencillo de aprender.

Los diversos aspectos de lo que se ha descrito anteriormente se pueden usar solos o en diversas combinaciones. La enseñanza de la presente solicitud se puede poner en práctica mediante una combinación de hardware y software, pero también se puede poner en práctica en hardware o software. La enseñanza de la presente solicitud también se puede materializar como código legible por ordenador en un medio legible por ordenador. Se debería hacer notar que la enseñanza de la presente solicitud no se limita del uso en terminales de comunicaciones móviles tales como teléfonos móviles, sino que se puede aplicar igualmente bien en Asistentes Digitales Personales (PDA), reproductores de MP3, organizadores personales o cualquier otro dispositivo diseñado para proporcionar información al tiempo que se mantiene un bajo consumo de energía.

La enseñanza de la presente solicitud tiene numerosas ventajas. Diferentes realizaciones o puestas en práctica pueden producir una o más de las siguientes ventajas. Se debería hacer notar que esta no es una lista exhaustiva y puede haber otras ventajas que no se describan en el presente documento. Una ventaja de la enseñanza de la presente solicitud es que se puede hacer que un dispositivo visualice información de una forma que es sencilla de aprender e intuitiva de usar.

Otra ventaja de la enseñanza de la presente solicitud es que un usuario puede operar determinadas funciones tales como recuperar información a partir de un dispositivo con únicamente una orden o movimiento sencillo e impreciso que no requiera la plena atención de un usuario.

Aunque la enseñanza de la presente solicitud se ha descrito con detalle para fines de ilustración, se entiende que tal detalle es únicamente para ese fin, y los expertos en la materia pueden hacer variaciones en la misma sin apartarse del alcance de la enseñanza de la presente solicitud.

Por ejemplo, aunque la enseñanza de la presente solicitud se ha descrito en términos de un teléfono móvil, se debería apreciar que las enseñanzas de la presente solicitud también se pueden aplicar a otros tipos de dispositivos electrónicos, tales como reproductores de música, ordenadores de bolsillo y similares. También se debería hacer notar que hay muchas formas alternativas de poner en práctica los métodos y aparatos de las enseñanzas de la presente solicitud.

El término "comprender", tal como se usa en las reivindicaciones, no excluye otros elementos o etapas. El término "un" o "una", tal como se usa en las reivindicaciones, no excluye una pluralidad. Una unidad u otros medios pueden cumplir las funciones de varias unidades o medios enumerados en las reivindicaciones.

TABLAS

Tabla 1: Sustituciones de coma flotante a coma fija.

Operación de coma flotante	Operación de coma fija
$x*y$	$(x*y)>>Q$
x/y	$(x<<Q)/y$
$\text{sqrt}(x)$	$\text{sqrt}(x<<Q)$
$\text{sen}(x)$	$\text{sen } T[(x>>Q)$
$\text{cos}(x)$	$\text{sen } T[(x>>Q) + 90]$

Tabla 2: Características de objeto y fines correspondientes.

Característica	Fin
Extensión	Para determinar la representación gráfica y para la detección de colisiones.
Posición	Para determinar la representación gráfica y para la detección de colisiones.
Masa	Para determinar el movimiento a través de aceleración, desaceleración, inercia y efecto tras una colisión.

ES 2 690 648 T3

Elasticidad	El comportamiento del objeto tras una colisión tanto para un movimiento continuado y como para la apariencia gráfica.
Rozamiento	Para determinar un movimiento continuado tras una colisión.
Valor de datos	Usado para notificar o portar un valor.
Función	Posibilita que una función se inicie tras la activación a través de un patrón de movimiento o una colisión.
Coefficiente elástico	Para determinar el movimiento.
Coefficiente de amortiguación	Para determinar el movimiento.
Velocidad	Para determinar el movimiento.
Aceleración	Para determinar el movimiento. Hay un valor implícito que también puede ser manejado exclusivamente por el controlador.
Fuerza	Para determinar el movimiento. Hay un valor implícito que también puede ser manejado exclusivamente por el controlador.
Orientación	Determinar qué mostrar en la representación gráfica y cómo determinar el movimiento adicional.

Tabla 3 Funciones y sus patrones de movimiento asociados.

Función	Patrón de movimiento
Para un recordatorio de llamada:	
Elevar	ABRIR
Desechar	SUPRIMIR
Colisionar con el símbolo de teléfono	Establecer llamada
Colisionar con la papelera	Suprimir
Colisionar con el símbolo de correo	Enviar mensaje
Elevación repetida	Abrir el siguiente recordatorio
Para una alarma	
Desechar	Posponer - la aceleración determina cuánto se retardará la alarma.
Golpecito	Repetición
Doble golpecito	Iniciar reproductor de medios y repetición.

REIVINDICACIONES

1. Un dispositivo (100) que comprende:

5 unos medios de detección de movimiento para detectar un movimiento; y
 unos medios de control para:

10 controlar unos medios de visualización para visualizar un movimiento emulado de una representación gráfica de un primer objeto (409) de acuerdo con un movimiento detectado por dichos medios de detección de movimiento, en donde dicho primer objeto (409) tiene un valor de datos y comprende una característica física; controlar unos medios de visualización para visualizar un segundo objeto (407) que tiene una función asociada, en donde la función es cualquiera tomada de entre el siguiente grupo:
 una operación de suprimir, una operación de llamar, una operación de enviar mensaje, una operación de posponer, una operación de abrir y una operación de leer en voz alta;
 15 generar dicho movimiento emulado sobre la base de dicha característica física, en donde los medios de control están configurados para, cuando un patrón de movimiento se corresponde con una colisión emulada entre el primer objeto (409) y el segundo objeto (407), ejecutar la función sobre el valor de datos tras la detección de dicha colisión emulada.

20 2. Un dispositivo (100) de acuerdo con la reivindicación 1, en donde dicha generación de dicho movimiento emulado comprende determinar una posición actualizada a partir de una posición original dependiendo de dicha característica física y dicho movimiento detectado por los medios de detección de movimiento y en donde dicho control de visualización de dicho movimiento emulado comprende controlar la visualización de dicho objeto en dicha posición actualizada.

25 3. Un dispositivo (100) de acuerdo con la reivindicación 1, que comprende adicionalmente unos medios para ejecutar una función asociada a un patrón de movimiento sobre dicho objeto tras la detección de dicho patrón de movimiento.

30 4. Un dispositivo (100) de acuerdo con la reivindicación 3, en donde dicho objeto representa una alarma, dicho patrón de movimiento se corresponde con un movimiento de lanzamiento y dicha función es una operación para posponer dicha alarma.

35 5. Un dispositivo (100) de acuerdo con la reivindicación 1, en donde dicho dispositivo (100) comprende adicionalmente unos medios para determinar si dicho movimiento emulado da como resultado una colisión entre dicho objeto y un objeto adicional.

40 6. Un dispositivo (100) de acuerdo con la reivindicación 5, en donde dicho objeto representa un contacto y dicho objeto adicional está asociado a una función para llamar a un contacto o dicho objeto adicional está asociado a una función para suprimir un elemento y en donde dicho dispositivo (100) comprende adicionalmente unos medios para ejecutar la función asociada tras determinar una colisión entre el objeto y el objeto adicional.

7. Un dispositivo electrónico que comprende un dispositivo según al menos una de las reivindicaciones 1 a 6 y unos medios de visualización.

45 8. Un método que comprende:

50 detectar un movimiento de un dispositivo (100),
 emular un movimiento para al menos un objeto de acuerdo con dicho movimiento, en donde dicho primer objeto (409) tiene un valor de datos y comprende una característica física y en donde dicha emulación se basa en dicha característica física;
 controlar la visualización del movimiento emulado del primer objeto (409);
 controlar la visualización de un segundo objeto (407) que tiene una función asociada, en donde la función es cualquiera tomada de entre el siguiente grupo: una operación de suprimir, una operación de llamar, una operación de enviar mensaje, una operación de posponer, una operación de abrir y una operación de leer en voz alta;
 55 en donde un patrón de movimiento se corresponde con una colisión emulada entre el primer objeto (409) y el segundo objeto (407) y el método comprende ejecutar dicha función sobre dicho valor de datos tras la detección de dicha colisión emulada.

60 9. Un método de acuerdo con la reivindicación 8,
 en donde un objeto tiene una función asociada que se corresponde con un patrón de movimiento y dicho método comprende adicionalmente
 determinar si dicho movimiento detectado completa dicho patrón de movimiento y, si es así, ejecutar dicha función.

65

10. Un método de acuerdo con la reivindicación 9, en donde dicho objeto representa una alarma, dicho patrón de movimiento se corresponde con un movimiento de lanzamiento y dicha función es una operación para posponer dicha alarma.

5 11. Un método de acuerdo con la reivindicación 8, que comprende adicionalmente determinar si dicho movimiento emulado da como resultado una colisión entre dicho objeto y un objeto adicional.

10 12. Un método de acuerdo con la reivindicación 11, en donde dicho objeto representa un contacto y dicho objeto adicional está asociado a una función para llamar a un contacto o dicho objeto adicional está asociado a una función para suprimir un elemento y en donde dicho método comprende adicionalmente ejecutar la función asociada tras determinar una colisión entre el objeto y el objeto adicional.

15 13. Un medio legible por ordenador que incluye al menos un código de programa informático para controlar un dispositivo (100) para realizar un método de acuerdo con al menos una de las reivindicaciones 8 a 12.

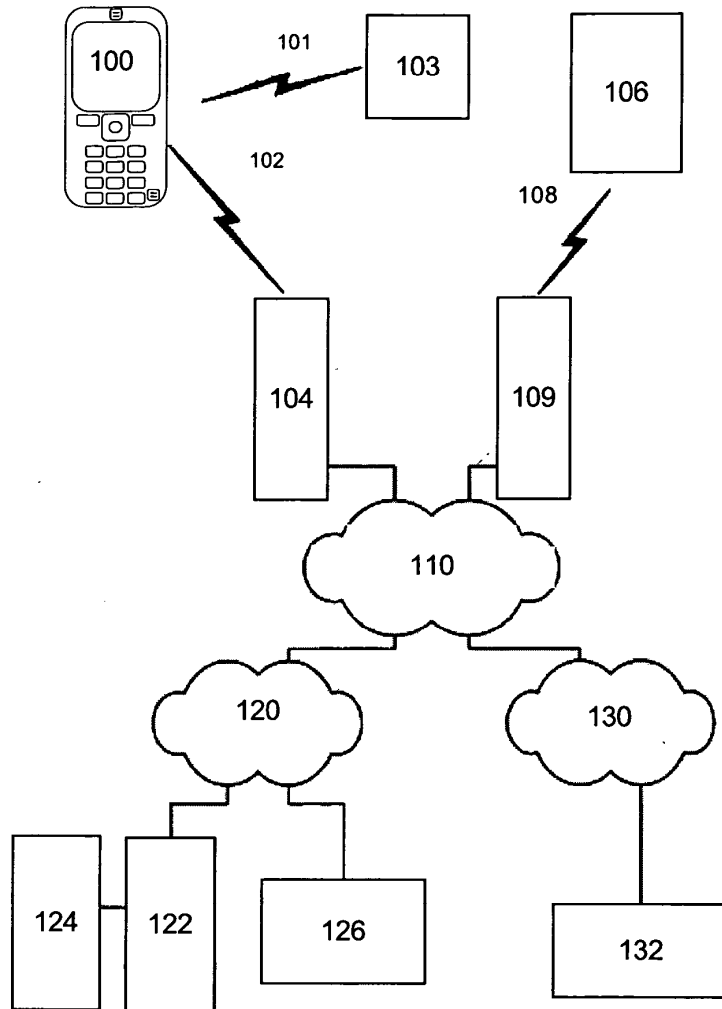


Fig 1

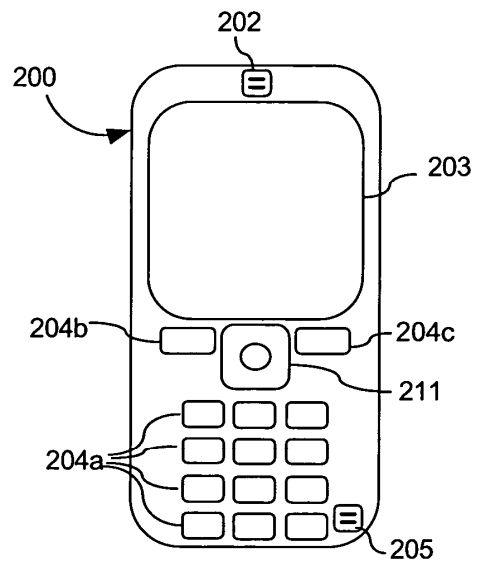


Fig 2

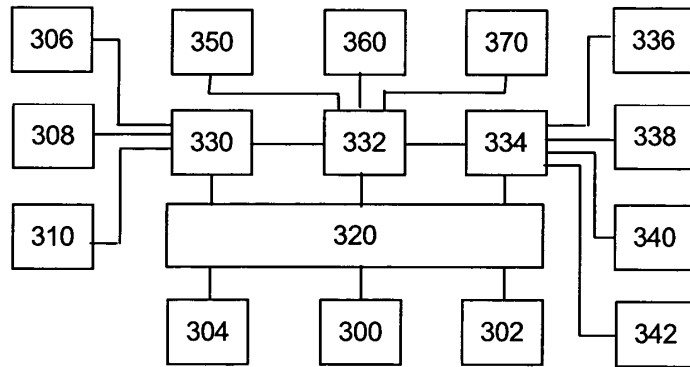


Fig 3

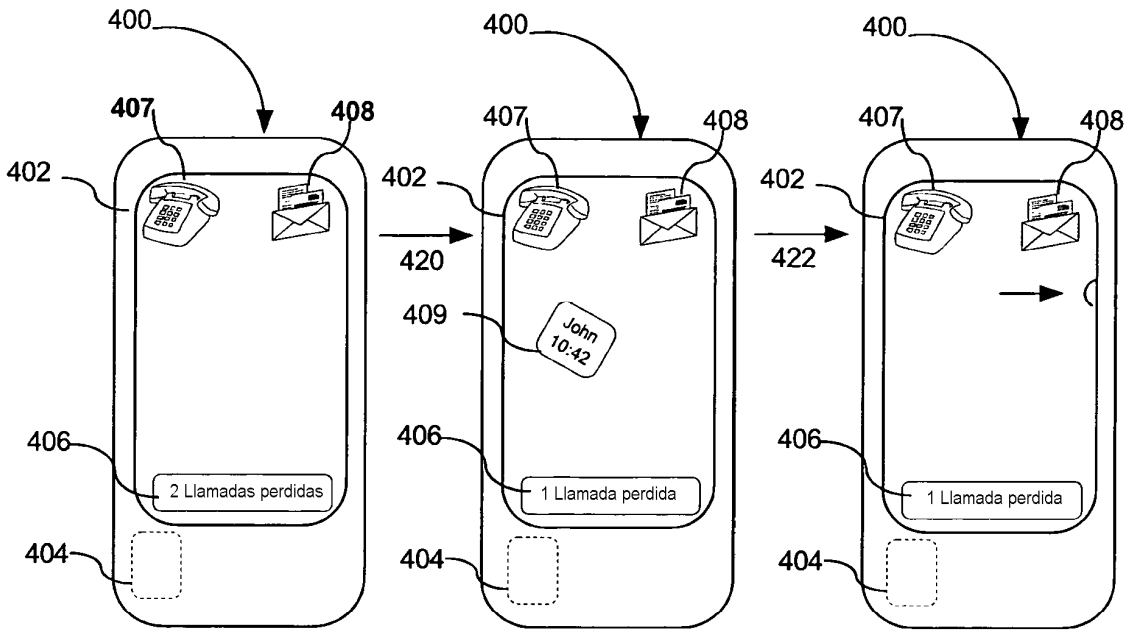


Fig 4a

Fig 4b

Fig 4c

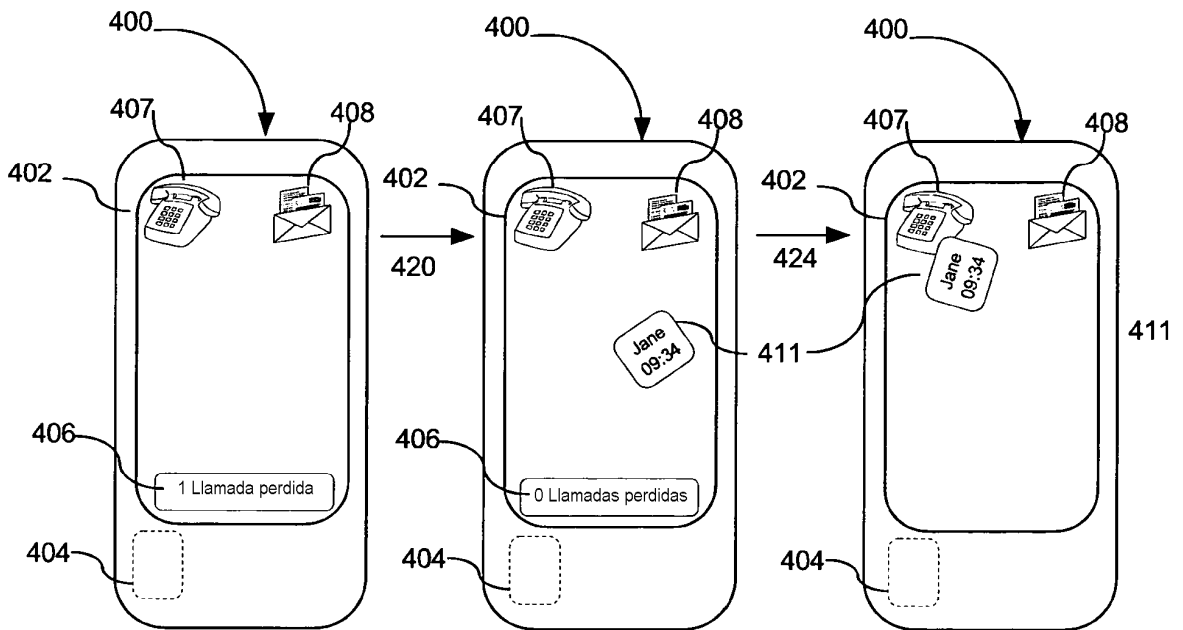


Fig 4d

Fig 4e

Fig 4f

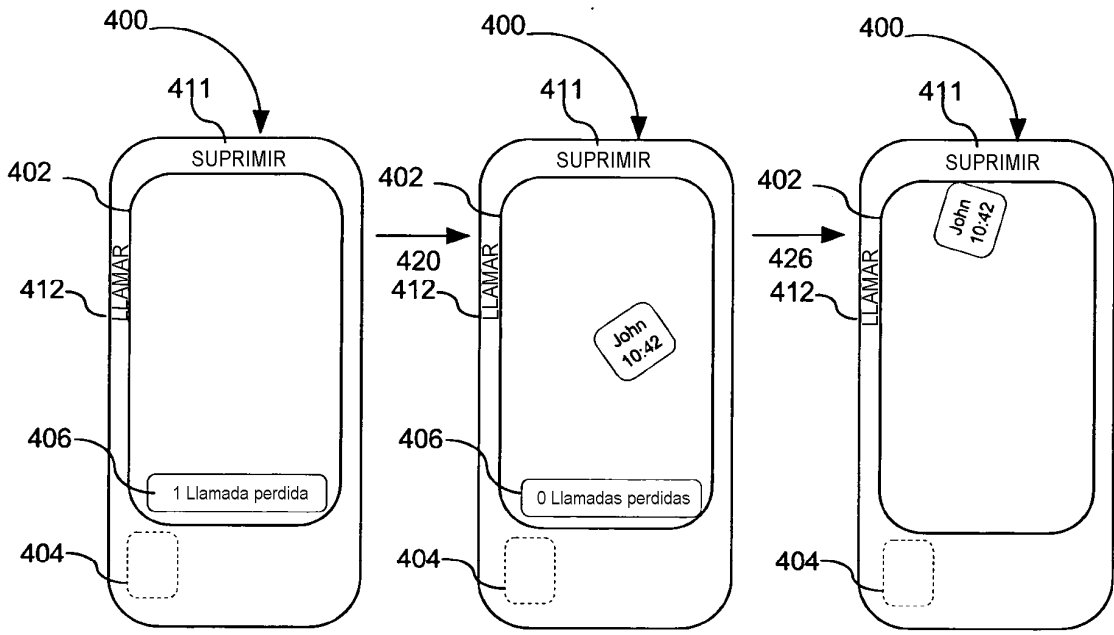


Fig 4g

Fig 4h

Fig 4i

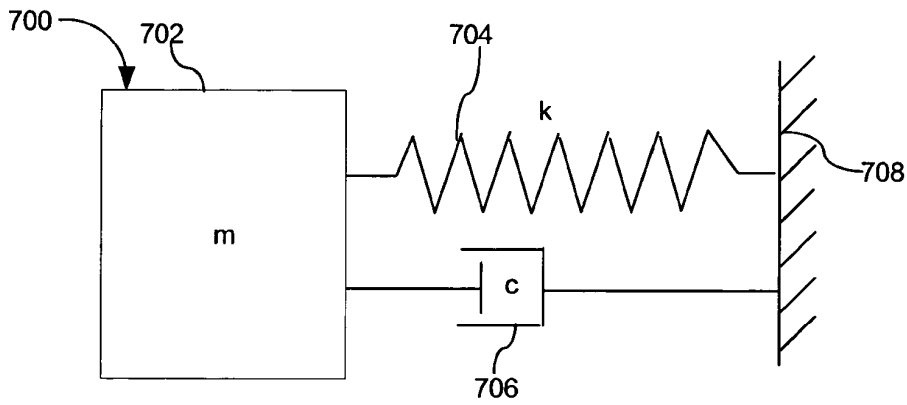


Fig 7a

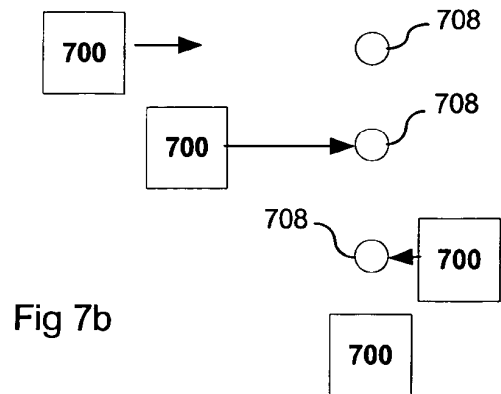


Fig 7b

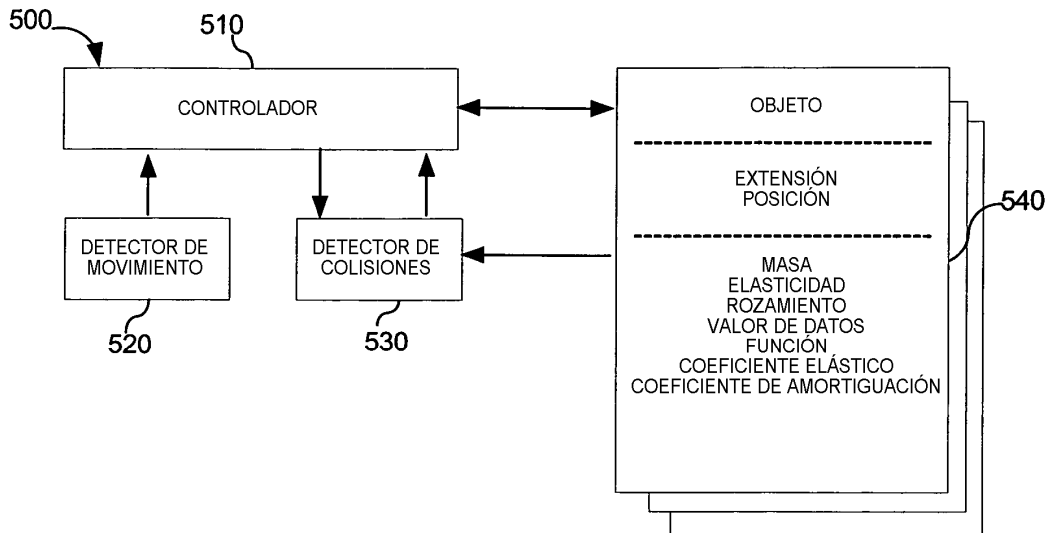


Fig 5a

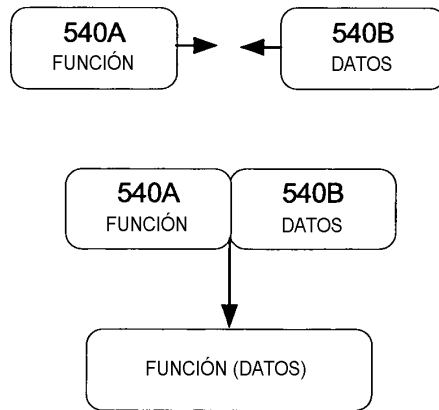


Fig 5b

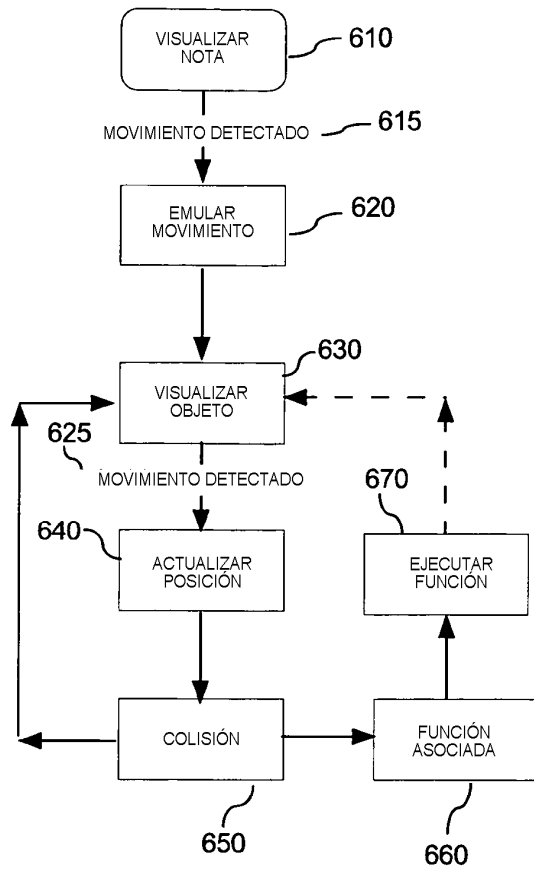


Fig 6a

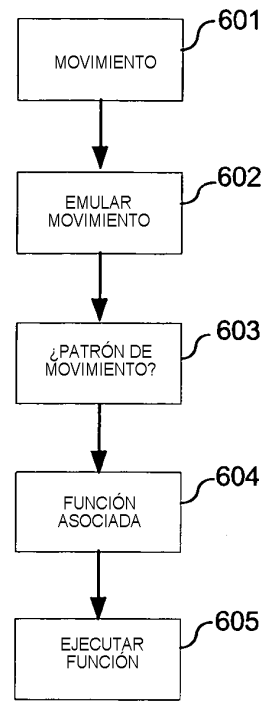


Fig 6b

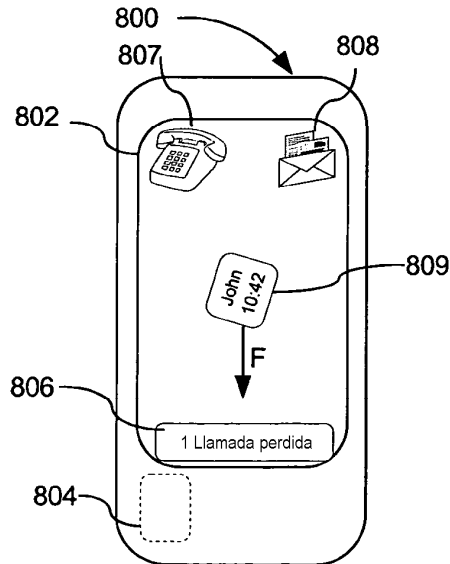


Fig 8