

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2011-54213

(P2011-54213A)

(43) 公開日 平成23年3月17日(2011.3.17)

(51) Int.Cl. F I テーマコード(参考)
G06F 3/048 (2006.01) G O 6 F 3/048 6 5 7 A 5 E 5 0 1
 G O 6 F 3/048 6 5 6 A

審査請求 有 請求項の数 6 O L (全 13 頁)

| | | | |
|------------|-----------------------------------|----------|------------------------|
| (21) 出願番号 | 特願2010-278042 (P2010-278042) | (71) 出願人 | 000003078 株式会社東芝 |
| (22) 出願日 | 平成22年12月14日 (2010.12.14) | | 東京都港区芝浦一丁目1番1号 |
| (62) 分割の表示 | 特願2009-66642 (P2009-66642) の分割 | (74) 代理人 | 100108855 弁理士 蔵田 昌俊 |
| 原出願日 | 平成21年3月18日 (2009.3.18) | (74) 代理人 | 100091351 弁理士 河野 哲 |
| | | (74) 代理人 | 100088683 弁理士 中村 誠 |
| | | (74) 代理人 | 100109830 弁理士 福原 淑弘 |
| | | (74) 代理人 | 100075672 弁理士 峰 隆司 |
| | | (74) 代理人 | 100095441 弁理士 白根 俊郎 |

最終頁に続く

(54) 【発明の名称】 情報処理装置および制御方法

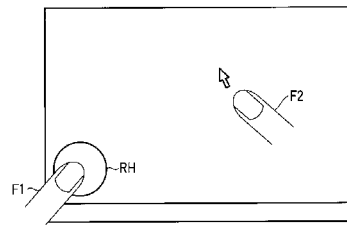
(57) 【要約】

【課題】 ホバリングモードであるか否かをユーザが確実に認識できるようにすること。

【解決手段】 実施形態に係る情報処理装置は、表示装置の表示画面に第1画像を表示させる第1制御手段と、前記表示画面に対するタッチ操作を受ける操作受付手段と、前記操作受付手段が、前記表示画面の一部の領域に位置する第1領域に対するタッチ操作を受けている間に前記第1画像に対するタッチ操作を受けた場合、前記第1画像についての情報を前記表示画面に表示させる第2制御手段と、前記操作受付手段が、前記第1領域に対するタッチ操作を受けていない間に前記第1画像に対するタッチ操作を受けた場合、前記第1画像に応じたアプリケーションを起動する第3制御手段とを備える。

【選択図】 図2

図2



【特許請求の範囲】**【請求項 1】**

表示装置の表示画面に第 1 画像を表示させる第 1 制御手段と、
前記表示画面に対するタッチ操作を受ける操作受付手段と、
前記操作受付手段が、前記表示画面の一部の領域に位置する第 1 領域に対するタッチ操作を受けている間に前記第 1 画像に対するタッチ操作を受けた場合、前記第 1 画像についての情報を前記表示画面に表示させる第 2 制御手段と、
前記操作受付手段が、前記第 1 領域に対するタッチ操作を受けていない間に前記第 1 画像に対するタッチ操作を受けた場合、前記第 1 画像に応じたアプリケーションを起動する第 3 制御手段と
を備える情報処理装置。

10

【請求項 2】

前記第 1 制御手段は、前記表示画面の前記第 1 領域の位置にボタンを表示させる、請求項 1 記載の情報処理装置。

【請求項 3】

前記第 1 領域は、前記表示画面における端部に位置する、請求項 1 記載の情報処理装置。

【請求項 4】

前記第 1 領域は、前記表示画面における左下部に位置する、請求項 1 記載の情報処理装置。

20

【請求項 5】

表示装置の表示画面に画像を表示させる表示制御手段と、
前記表示画面に対するタッチ操作を受ける操作受付手段と、
前記操作受付手段が、前記表示画面の一部の領域に位置する第 1 領域に対するタッチ操作を受けているか否かに応じて、ホバリングモードと通常モードとを切り替える切替手段と
を備える情報処理装置。

【請求項 6】

表示装置に画像を表示させる情報処理装置における制御方法であって、
前記表示装置の表示画面に第 1 画像を表示することと、
前記表示画面の一部の領域に位置する第 1 領域に対するタッチ操作を受けている間に前記第 1 画像に対するタッチ操作を受けた場合、前記第 1 画像についての情報を表示することと、
前記第 1 領域に対するタッチ操作を受けていない間に前記第 1 画像に対するタッチ操作を受けた場合、前記第 1 画像に応じたアプリケーションを起動することと
を備える制御方法。

30

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、ホバリングモードと通常のモードとを切り替える情報処理装置および制御方法に関する。

40

【背景技術】**【0002】**

近年、表示装置と、表示装置の表示画面上に設けられた位置検出装置とを備えたタッチパネルを備えたノートブック型パーソナルコンピュータが販売されている。

【0003】

位置検出装置には、抵抗膜方式や電磁誘導方式などの方式がある。一般に、位置検出装置が検出した位置に表示されているウィンドウを制御するアプリケーションプログラムに位置を示すイベントとマウスボタンが押下されていることを示すイベントを渡す。

【0004】

50

しかし、常にマウスボタンが押下された状態であるとアプリケーションプログラムが認識していると不便であるため、ホバリングモードというモードを設けることがある。ホバリングモードでは、アプリケーションプログラムに位置を示すイベントのみを渡す。

【0005】

特許文献1は、表示画面にホバリングモードに設定するためのホバリングアイコンを設け、通常モードでホバリングアイコンにタッチすると、次にホバリングアイコンにタッチするまでホバリングモードに設定することを開示している。

【先行技術文献】

【特許文献】

【0006】

【特許文献1】特開2000-122808号公報(段落番号[0022])

【発明の概要】

【発明が解決しようとする課題】

【0007】

上述した文献が開示する技術では、ホバリングアイコンへのタッチすることで、ホバリングモードと通常モードとを切り替えている。従って、コンピュータの前から暫く席を外していたりすると、現在のモードがホバリングモードと通常モードとのどちらであるかをユーザが分からなくなることがある。

【0008】

本発明の目的は、ホバリングモードで有るか否かをユーザが確実に認識することが可能な情報処理装置および制御方法を提供することにある。

【課題を解決するための手段】

【0009】

本発明の一例に係わる情報処理装置は、表示装置の表示画面に第1画像を表示させる第1制御手段と、前記表示画面に対するタッチ操作を受ける操作受付手段と、前記操作受付手段が、前記表示画面の一部の領域に位置する第1領域に対するタッチ操作を受けている間に前記第1画像に対するタッチ操作を受けた場合、前記第1画像についての情報を前記表示画面に表示させる第2制御手段と、前記操作受付手段が、前記第1領域に対するタッチ操作を受けていない間に前記第1画像に対するタッチ操作を受けた場合、前記第1画像に応じたアプリケーションを起動する第3制御手段とを備える。

【発明の効果】

【0010】

本発明によれば、ホバリングモードと通常モードのどちらで有るかをユーザが確実に認識することが可能になる。

【図面の簡単な説明】

【0011】

【図1】第1の実施形態に係わる情報処理装置のシステム構成を示すブロック図。

【図2】ホバリングモード領域内を指で押圧することによってホバリングモードに設定した状態で、別の箇所を指で押圧している状態を示す図。

【図3】第1の実施形態に係わるホバリングモード/ノーマルモードの切り替え、およびホバリングモードに設定されている場合にボタンダウンイベントをマウスカーソルが描画されているアプリケーションプログラムに渡さないための構成を示す図。

【図4】第1の実施形態に係わるホバリングモードおよびノーマルモードに設定するためのモード設定モジュールが処理する手順を示すフローチャート。

【図5】第1の実施形態に係わるマウスイベントが渡されたときのイベントハンドルモジュールの処理の手順を示すフローチャート。

【図6】ホバリングモードに設定することでツールチップを表示させている状態を示す図。

【図7】ホバリングモードに設定することでランチャプログラムが吹き出しを表示している状態を示す図。

10

20

30

40

50

【図 8】ホバリングモードに設定することでマウスジェスチャを行っている状態を示す図。

【図 9】ユーザがボタンを押下してホバリングモードに設定した状態で、表示画面を指で押圧している状態を示す図。

【図 10】第 2 の実施形態に係わるホバリングモード / ノーマルモードの切り替え、およびホバリングモードに設定されている場合にボタンダウンイベントをマウスカーソルが描画されているアプリケーションプログラムに渡さないための構成を示す図。

【図 11】第 2 の実施形態に係わるホバリングモードおよびノーマルモードに設定するためのモード設定モジュールが処理する手順を示すフローチャート。

【図 12】第 2 の実施形態に係わるマウスイベントが渡されたときのイベントハンドルモジュールの処理の手順を示すフローチャート。

【発明を実施するための形態】

【0012】

本発明の実施の形態を以下に図面を参照して説明する。

【0013】

(第 1 の実施形態)

まず、図 1 を参照して、本発明の一実施形態に係る情報処理装置の構成について説明する。この情報処理装置は、バッテリー駆動可能な携帯型のノートブック型パーソナルコンピュータとして実現されている。

【0014】

次に、図 1 を参照して、本コンピュータのシステム構成について説明する。

【0015】

本コンピュータは、図 1 に示されているように、ディスプレイユニット 12、CPU 111、ノースブリッジ 112、主メモリ 113、グラフィクスコントローラ 114、サウスブリッジ 119、BIOS-ROM 120、ハードディスクドライブ 121、エンベデッドコントローラ / キーボードコントローラ IC 124、および電源コントローラ 125 等を備えている。

【0016】

CPU 111 は、本コンピュータの動作を制御するために設けられたプロセッサであり、ハードディスクドライブ 121 から主メモリ 113 にロードされる、オペレーティングシステム、および各種アプリケーションプログラムを実行する。

【0017】

また、CPU 111 は、BIOS-ROM 120 に格納された BIOS プログラムも実行する。BIOS プログラムはハードウェア制御のためのプログラムである。

【0018】

ノースブリッジ 112 は CPU 111 のローカルバスとサウスブリッジ 119 との間を接続するブリッジデバイスである。ノースブリッジ 112 には、主メモリ 113 をアクセス制御するメモリコントローラも内蔵されている。また、ノースブリッジ 112 には、AGP バスなどを介してグラフィクスコントローラ 114 との通信を実行する機能も有している。

【0019】

ディスプレイユニット 12 には、LCD 17 から構成される表示装置が組み込まれている。LCD 17 の画像表示面上に、感圧式 (抵抗膜方式) のタッチセンサ 18 が装着されている。タッチセンサ 18 は、表面が押圧された場合に、押圧箇所を示す検出信号を出力する。本実施形態のタッチセンサ 18 は、同時に 2 箇所の押圧箇所を検出することができ、検出したそれぞれの押圧箇所を示す二つの検出信号を出力することが可能である。

【0020】

グラフィクスコントローラ 114 は、本コンピュータのディスプレイモニタとして使用される LCD 17 を制御する表示コントローラである。このグラフィクスコントローラ 114 はビデオメモリ 114A を有しており、OS / アプリケーションプログラムによって

10

20

30

40

50

ビデオメモリ 114A に書き込まれた表示データから、ディスプレイユニット 12 の LCD 17 に表示すべき表示イメージを形成する映像信号を生成する。

【0021】

サウスブリッジ 119 は、LPC バス上のデバイスを制御する。また、サウスブリッジ 119 は、HDD 121 を制御するための IDE コントローラを内蔵している。さらに、サウスブリッジ 119 は、および BIOS-ROM 120 をアクセス制御するための機能も有している。また、更にサウスブリッジ 119 は、USB 規格をサポートした機器とのデータの伝送の制御を行うための USB コントローラ 119A を有している。USB 規格をサポートした機器であるタッチセンサ 18 は、サウスブリッジ 119 内の USB コントローラ 119A とデータの伝送を行う。

10

【0022】

エンベデッドコントローラ/キーボードコントローラ IC 124 は、電力管理のためのエンベデッドコントローラと、キーボード 13、タッチパッド 16 を制御するためのキーボードコントローラとが集積された 1 チップマイクロコンピュータである。このエンベデッドコントローラ/キーボードコントローラ IC 124 は、電源コントローラ 125 と協調して動作することにより、ユーザによるパワーボタン 14 の操作に応じて、本コンピュータをパワーオン/パワーオフする機能を有している。

【0023】

タッチセンサ 18 が検出信号を出力している場合、オペレーティングシステムは、ポインティングデバイスのボタン操作に係わるイベントとしてマウスイベントを発行する。この時、オペレーティングシステムが発行するマウスイベントは、マウスカーソルの位置を示す位置イベントとマウスボタンが押されていることを示すボタンダウンイベントである。

20

【0024】

通常のマウスで操作を行った場合、エクスプローラやデスクトップのアイコン上にポインタを重ねると、アイコンが示しているファイル等の補足情報を示すツールチップが表示される。

【0025】

ところが、タッチパネルでポインタを操作した場合、以下に示すような不都合が生じる。タッチパネルでポインタを操作すると常にボタンダウンイベントが発行されているために、ツールチップを表示することができない。

30

【0026】

本装置では、このような不都合が生じることを防止することが可能である。以下に、不都合の発生を防止するための構成について説明する。

図 2 に示すように、ディスプレイの例えば左下隅の領域を仮想的なボタンであるホバリングモード領域 RH を設定する。そして、このホバリングモード領域 RH 内を指 F1 等で押圧していることを示す検出信号がタッチセンサから出力されている場合に、ホバリングモードになる。ホバリングモードでは、ホバリングモード領域 RH 以外の領域を指 F2 等で押圧していることを示す検出信号がタッチセンサから出力されている場合に、マウスボタンの操作に係わるイベントをマウスカーソルが描画されているアプリケーションプログラムに渡さずに、マウスカーソルの位置を示すイベントのみをマウスカーソルが描画されているアプリケーションプログラムを渡す。ホバリングモード領域 RH 以外の領域を指 F2 等で押圧していない場合はノーマルモードである。ノーマルモードでは、マウスボタンの操作およびマウスカーソルの位置を示すイベントをマウスカーソルが描画されているアプリケーションプログラムを渡す。

40

【0027】

図 3 を参照して、ホバリングモード/ノーマルモードの切り替え、およびホバリングモードに設定されている場合にボタンダウンイベントをマウスカーソルが描画されているアプリケーションプログラムに渡さないための構成を説明する。

【0028】

50

図3に示すように、オペレーティングシステム201、デバイスドライバ202、イベントフック/ハンドルモジュール210等から構成されている。

【0029】

デバイスドライバ202は、通常のタッチセンサ18のデバイスドライバと同じドライバを用いる。従って、デバイスドライバ202は、タッチセンサ18が検出信号から押圧箇所を示す位置情報を取得する。デバイスドライバ202は、位置情報をオペレーティングシステム201に渡す。オペレーティングシステム201は、位置情報が示している位置に表示されているアプリケーションプログラム(デスクトップ環境アプリケーション221、またはランチャアプリケーション222)に位置イベントおよびボタндаウンイベントを含むマウスイベントを渡すための処理を行う。なお、デスクトップ環境アプリケーション221は、表示画面にウィンドウの一つとしてデスクトップ画面を表示するウィンドウ制御手段である。また、ランチャアプリケーション222は、デスクトップ画面上にアプリケーションを起動するためのボタンを備えたウィンドウを表示するウィンドウ制御手段である。

10

【0030】

イベントフック/ハンドルモジュール210は、フックモジュール211、モード設定モジュール212、およびイベントハンドルモジュール213等を備えている。フックモジュール211は、デバイスドライバ202から出力された位置イベントおよびボタндаウンイベントをフックする。モード設定モジュール212は、位置イベントが示す表示画面上の位置に基づいて、フックモジュール211は、ホバリングモードとノーマルモードとの切り替えを行う。イベントハンドルモジュール213は、ノーマルモードの場合に位置イベントおよびボタндаウンイベントを元々の送り先であるアプリケーションプログラムに渡す。また、イベントハンドルモジュール213は、ホバリングモードの場合に位置イベントのみを元々の送り先であるアプリケーションプログラムに渡す。

20

【0031】

次に、図4のフローチャートを参照してホバリングモードおよびノーマルモードに設定するためのモード設定モジュール212が処理する手順について説明する。

まず、フックモジュール211が位置イベントおよびボタндаウンイベントをフックする(ステップS10)。モード設定モジュール212は、二つのマウスイベントをフックしたか否かを判断する(ステップS11)。二つのマウスイベントをフックしていないと判断した場合、(ステップS11のNo)、モード設定モジュール212はマウスイベントの位置イベントから表示画面上の位置を示す位置情報を取得する(ステップS12)。モード設定モジュール212は、位置情報がホバリングモード領域RH内を示しているか否かを判別する(ステップS13)。ホバリングモード領域RH内を示していると判断した場合(ステップS13のYes)、フラグ214の値をTrueにすることによってホバリングモードに設定する(ステップS14)。そして、マウスイベントをイベントハンドルモジュール213に渡す(ステップS15)。ステップS13において、ホバリングモード領域RH内を示していないと判断した場合(ステップS13のNo)、モード設定モジュール212はフラグ214の値をFalseにすることによってノーマルモードに設定する(ステップS16)。そして、マウスイベントをイベントハンドルモジュール213に渡す(ステップS17)。

30

40

【0032】

ステップS11において、二つのマウスイベントをフックしたと判断した場合、(ステップS11のYes)、二つのマウスイベントの一方のイベント(以後第1マウスイベントとする)から表示画面上の位置を示す位置情報を取得する(ステップS22)。モード設定モジュール212は、位置情報がホバリングモード領域RH内を示しているか否かを判別する(ステップS23)。ホバリングモード領域RH内を示していると判断した場合(ステップS23のYes)、フラグ214の値をTrueにすることによってホバリングモードに設定する(ステップS24)。そして、二つのマウスイベントの他方のイベント(以後第2マウスイベントとする)をイベントハンドルモジュール213に渡す(ステ

50

ップ S 2 5)。

【 0 0 3 3 】

ステップ S 2 3 において、ホバリングモード領域 R H 内を示していないと判断した場合 (ステップ S 2 3 の N o)、第 2 マウスイベントから表示画面上の位置を示す位置情報を取得する (ステップ S 3 2)。モード設定モジュール 2 1 2 は、位置情報がホバリングモード領域 R H 内を示しているか否かを判別する (ステップ S 3 3)。ホバリングモード領域 R H 内を示していると判断した場合 (ステップ S 3 3 の Y e s)、フラグ 2 1 4 の値を T r u e にすることによってホバリングモードに設定する (ステップ S 3 4)。そして、第 2 マウスイベントをイベントハンドルモジュール 2 1 3 に渡す (ステップ S 3 5)。

【 0 0 3 4 】

ステップ S 3 3 において、ホバリングモード領域 R H 内を示していないと判断した場合 (ステップ S 3 3 の N o)、モード設定モジュール 2 1 2 はフラグ 2 1 4 の値を F a l s e にすることによってノーマルモードに設定する (ステップ S 3 6)。そして、マウスイベントをイベントハンドルモジュール 2 1 3 に渡す (ステップ S 3 7)。

【 0 0 3 5 】

フックモジュール 2 1 1 がマウスイベントを取得した場合の処理の手順の説明を終了する。以上の処理によって、ユーザがホバリングモード領域 R H 内を押圧している場合に、ホバリングモードに設定することができる。

【 0 0 3 6 】

ユーザがホバリングモード領域 R H 内を押圧している場合にホバリングモードに設定され、ユーザがホバリングモード領域 R H 内を押圧していない場合にノーマルモードに設定されるので、ユーザがどちらのモードで有るかを確実に識別することが可能になる。

【 0 0 3 7 】

次に、図 5 を参照して、マウスイベントが渡されたときのイベントハンドルモジュール 2 1 3 の処理の手順を説明する。

マウスイベントが渡されると、イベントハンドルモジュール 2 1 3 は二つのマウスイベントが渡されたか否かを判別する (ステップ S 4 1)。二つのマウスイベントが渡されていないと判断した場合 (ステップ S 4 1 の N o)、イベントハンドルモジュール 2 1 3 は、フラグ 2 1 4 の値を参照し、フラグ 2 1 4 の値 (T r u e 、 F a l s e) からホバリングモードで有るか否かを判別する (ステップ S 4 2)。ホバリングモードであると判断した場合 (ステップ S 4 2 の Y e s)、イベントハンドルモジュール 2 1 3 はマウスイベントからポタダウンイベントを削除し、位置イベントのみを残す (ステップ S 4 3)。そして、ポタダウンイベントが削除されたマウスイベントを、マウスイベントの元々の渡し先であるアプリケーションプログラムに渡す (ステップ S 4 4)。

【 0 0 3 8 】

ステップ S 4 2 においてホバリングモードではないと判断した場合 (ステップ S 4 2 の N o)、イベントハンドルモジュール 2 1 3 は、マウスイベントを変更せずに、マウスイベントの元々の渡し先であるマウスイベントをアプリケーションプログラムに渡す (ステップ S 4 5)。ステップ S 4 1 において、二つのマウスイベントが渡されたと判断した場合 (ステップ S 4 1 の Y e s)、イベントハンドルモジュール 2 1 3 は、それぞれのマウスイベントの元々の渡し先であるアプリケーションプログラムにそれぞれ渡す (ステップ S 4 5)。

【 0 0 3 9 】

以上の処理によって、ホバリングモードの場合に汎用のデバイスドライバが発行したマウスイベントをポタダウンイベントを削除したマウスイベントに変更することができる。

【 0 0 4 0 】

ホバリングモードに設定している場合、以下のようなメリットがある。

例えば、図 6 に示すように、表示画面上のホバリングモード領域 R H 上を指 F 1 で押圧してホバリングモードにした状態で、指 F 2 で押圧することによってデスクトップ画面のアイコン 3 0 1 にマウスカーソル 3 0 2 を重ねる。すると、アイコン 3 0 1 が示すファ

10

20

30

40

50

イルの情報を示すツールチップ 3 0 3 を表示することが可能になる。

【 0 0 4 1 】

また、図 7 に示すように、表示画面上のホバリングモード領域 R H 上を指 F 1 で押圧してホバリングモードにした状態で、指 F 2 で押圧することによってランチャプログラムが表示するウィンドウ 4 0 1 内のボタン 4 0 2 上にマウスカーソル 4 0 3 を重ねる。すると、ボタン 4 0 2 の操作によって起動されるアプリケーションプログラムの情報を吹き出し 4 0 4 に表示することができる。

【 0 0 4 2 】

また、図 8 に示すように、表示画面上のホバリングモード領域 R H 上を指 F 1 で押圧してホバリングモードにした状態で、指 F 2 で押圧しつつ移動させることによって、マウスカーソルを移動させることによってマウスジェスチャーモードを加えることにより、ホバリングによるマウスジェスチャーを行うことが出来る。なお、表示画面上に表示されている線 5 0 2 はマウスカーソル 5 0 1 の移動の軌跡である。

【 0 0 4 3 】

なお、デジタイザのようなタッチスクリーンでは、専用ペンを用いることでホバリングを実現していた。しかし本装置では、指によって、ペンでのホバリングと同様の操作を行うことができる。また、ウィンドウズ（登録商標）でマウス操作に慣れたユーザは、タッチでの“選択”が主な操作となるタッチディスプレイを使用する際にわずらわしさを感じると考えられる。ホバリング機能によって従来のマウスと同様の操作を実現すれば、わずらわしさから開放される。

【 0 0 4 4 】

（第 2 の実施形態）

なお、表示画面上のホバリングモード領域 R H 内を押圧するのではなく、図 9 に示すように、コンピュータ 1 0 の本体に設けられたユーザがホバリングモードを設定するための専用のボタン 6 0 1 を指 F 1 によって押下することによってホバリングモードに設定されるようにしても良い。なお、モード設定用のボタン 6 0 1 の代わりにキーボード 1 3 中の特定のキー（例えば、A l t キー）を押下することによってホバリングモードに設定されるようにしても良い。なお、図 9 において、コンピュータ 1 0 の表示画面を指 F 2 で押圧している。

【 0 0 4 5 】

ボタン 6 0 1 を用いてホバリングモードに設定するイベントフック / ハンドルモジュールの構成を図 1 0 に示す。

ボタン 6 0 1 は、ホバリングモードに設定するためのキーである。ユーザがボタン 6 0 1 を押下することによってホバリングモードに設定される。ボタン 6 0 1 を押下されていない場合にノーマルモードに設定される。

【 0 0 4 6 】

オペレーティングシステム 2 0 1 は、通常のキーのデバイスドライバと同じドライバを用いる。

【 0 0 4 7 】

図 1 0 に示すように、イベントフック / ハンドルモジュール 6 1 0 は、マウスイベントフックモジュール 6 1 1、マウスイベントハンドルモジュール 6 1 2、キーイベントフックモジュール 6 1 3、モード設定モジュール 6 1 4、およびフラグ 6 1 5 等を備えている。

【 0 0 4 8 】

マウスイベントフックモジュール 6 1 1 は、オペレーティングシステム 2 0 1 が、アプリケーションに対して発行したマウスイベントをフックする。マウスイベントハンドルモジュール 6 1 2 は、図 3 で説明したイベントハンドルモジュール 2 1 3 と同様に、ホバリングモードの場合に、マウスイベント中のボタンダウンイベントを削除し、位置イベントが含まれるマウスイベントを元々の発行先であるアプリケーションに渡す。

【 0 0 4 9 】

キーイベントフックモジュール613は、オペレーティングシステム201が発行したキーイベントをフックし、モード設定モジュール614に渡す。モード設定モジュール614は、渡されたキーイベントがボタン601の押下を示す場合に、フラグ615の値をTrueにしてホバリングモードを設定する。また、モード設定モジュール614は、渡されたキーイベントがボタン601の押下を示さない場合に、フラグ615の値をFalseにしてノーマルモードを設定する。また、モード設定モジュール614は、キーイベントが渡されていない場合に、フラグ615の値をFalseにしてノーマルモードを設定する。

【0050】

次に、図11のフローチャートを参照してホバリングモードおよびノーマルモードを設定する処理の手順を説明する。

まず、モード設定モジュール614は、キーイベントフックモジュール613からキーイベントを渡されたか否かを判別する(ステップS51)。キーイベントを渡されていないと判断した場合(ステップS51のNo)、フラグ615の値をFalseにしてノーマルモードに設定する(ステップS54)。キーイベントを渡されたと判断した場合(ステップS51のYes)、モード設定モジュール614は、キーイベントがホバリングモードに設定するためのボタン601の押下を示すキーイベントであるか否かを判別する(ステップS52)。ボタン601の押下を示していると判断した場合(ステップS52のYes)、フラグ615の値をTrueにしてホバリングモードに設定する(ステップS53)。ボタン601の押下を示していないと判断した場合(ステップS52のNo)、

【0051】

ユーザがボタン601を押下している場合にホバリングモードに設定され、ユーザがボタン601を押下していない場合にノーマルモードに設定されるので、ユーザがどちらのモードで有るかを確実に識別することが可能になる。

【0052】

次に、図12のフローチャートを参照してマウスイventフックモジュール513がマウスイventをフックした場合の処理の手順を説明する。

まず、マウスイventフックモジュール513が、マウスイventをフックする(ステップS61)。マウスイventフックモジュール513は、フックしたマウスイventをマウスイventハンドルモジュール612に渡す。

【0053】

マウスイventハンドルモジュール612は、フラグ615の値を参照し、フラグ615の値(True、False)からホバリングモードで有るか否かを判別する(ステップS62)。ホバリングモードであると判断した場合(ステップS62のYes)、マウスイventハンドルモジュール612はマウスイventからボタндаウンイベントを削除し、位置イベントのみを残す(ステップS63)。そして、マウスイventハンドルモジュール612は、ボタндаウンイベントが削除されたマウスイventを、マウスイventの元々の渡し先であるアプリケーションプログラムに渡す(ステップS64)。

【0054】

ホバリングモードではないと判断した場合(ステップS62のNo)、マウスイventハンドルモジュール612は、マウスイventを変更せずに、マウスイventの元々の渡し先であるマウスイventをアプリケーションプログラムに渡す(ステップS65)。

【0055】

以上の処理によって、ホバリングモードの場合に汎用のデバイスドライバが発行したマウスイventをボタндаウンイベントを削除したマウスイventに変更することができる。

【0056】

以上説明したように、表示画面内のホバリングモード領域内の押圧操作、またはボタン601の押下操作を行っている場合にホバリングモードに設定するようにすることによって、ユーザが現在ホバリングモードおよびノーマルモードの何れかであることを確実に識

10

20

30

40

50

別することが可能になる。

【 0 0 5 7 】

なお、本発明は、上記実施形態そのままに限定されるものではなく、実施段階ではその要旨を逸脱しない範囲で構成要素を変形して具体化できる。また、上記実施形態に開示されている複数の構成要素の適宜な組み合わせにより種々の発明を形成できる。例えば、実施形態に示される全構成要素から幾つかの構成要素を削除してもよい。更に、異なる実施形態に亘る構成要素を適宜組み合わせてもよい。

【 符号の説明 】

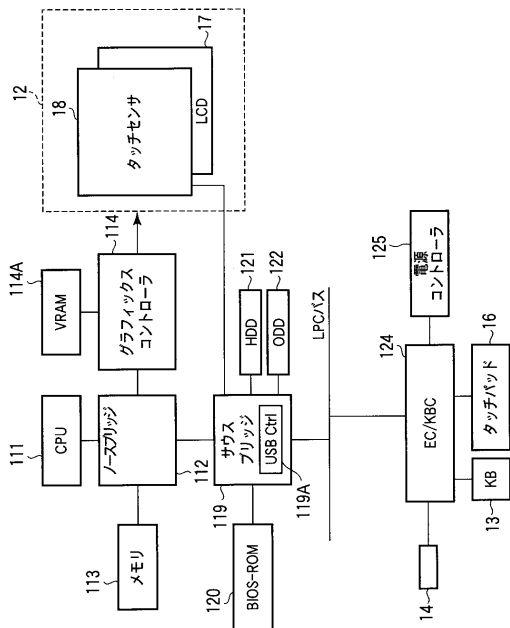
【 0 0 5 8 】

12 ... ディスプレイユニット, 17 ... LCD, 18 ... タッチセンサ, 201 ... オペレーティングシステム, 202 ... デバイスドライバ, 210 ... ハンドルモジュール, 211 ... フックモジュール, 211 ... フックモジュール, 212 ... モード設定モジュール, 213 ... イベントハンドルモジュール, 214 ... フラグ, 221 ... デスクトップ環境アプリケーション, 222 ... ランチャアプリケーション, RH ... ホバリングモード領域, 601 ... ボタン, 610 ... ハンドルモジュール, 611 ... マウスイベントフックモジュール, 612 ... マウスイベントハンドルモジュール, 613 ... キーイベントフックモジュール, 613 ... キーイベントフックモジュール, 614 ... モード設定モジュール, 614 ... モード設定モジュール, 615 ... フラグ。

10

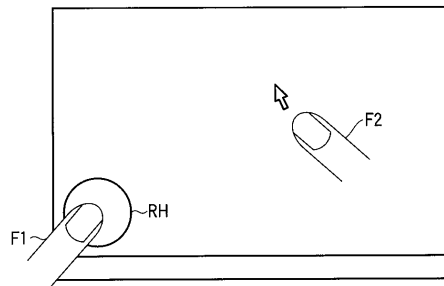
【 図 1 】

図 1

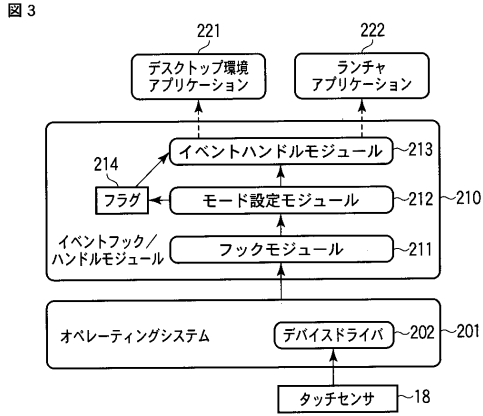


【 図 2 】

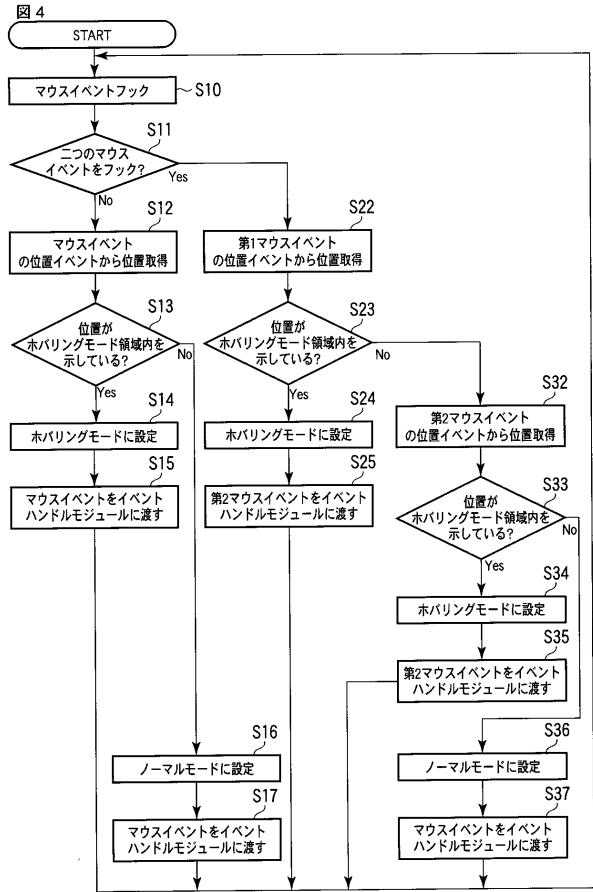
図 2



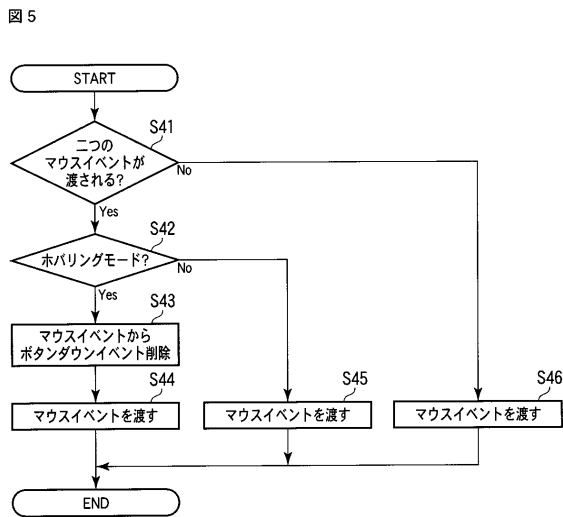
【 図 3 】



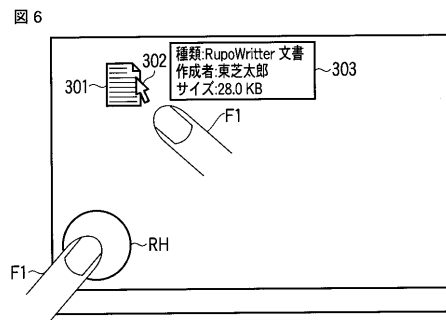
【 図 4 】



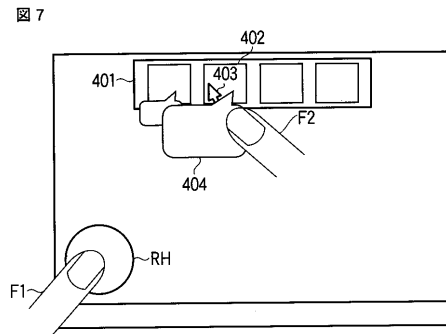
【 図 5 】



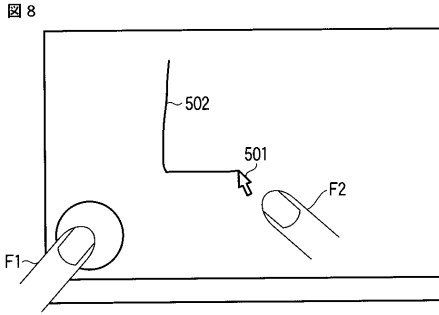
【 図 6 】



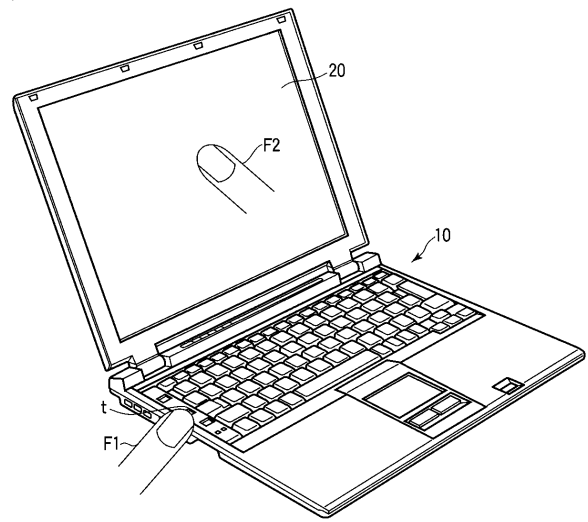
【 図 7 】



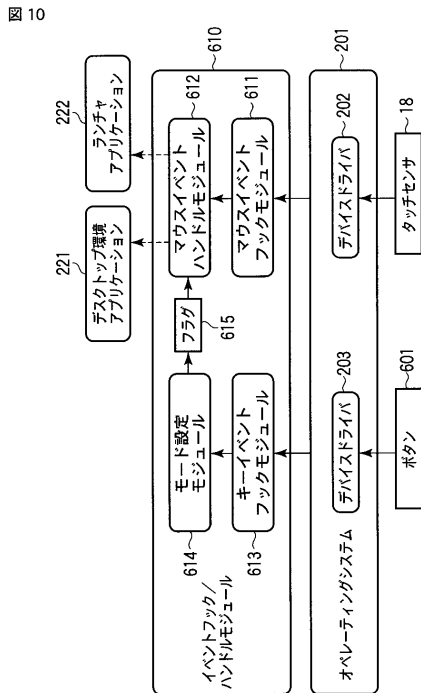
【 図 8 】



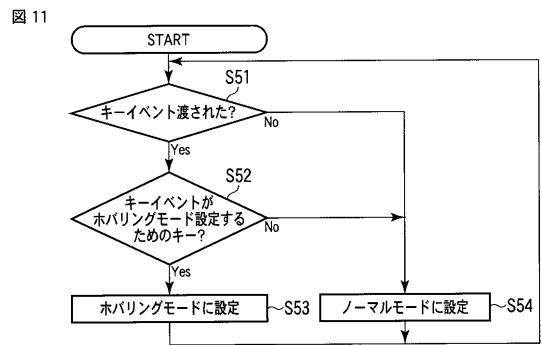
【 図 9 】



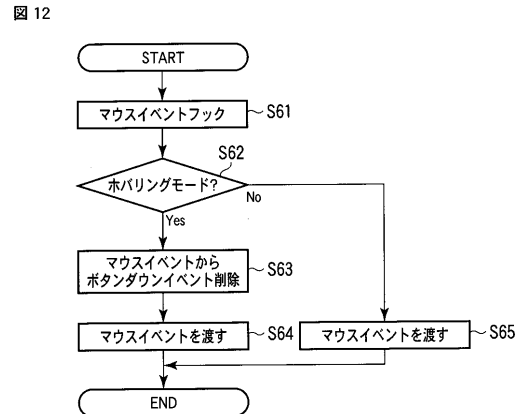
【 図 10 】



【 図 11 】



【 図 12 】



フロントページの続き

- (74)代理人 100084618
弁理士 村松 貞男
- (74)代理人 100103034
弁理士 野河 信久
- (74)代理人 100119976
弁理士 幸長 保次郎
- (74)代理人 100153051
弁理士 河野 直樹
- (74)代理人 100140176
弁理士 砂川 克
- (74)代理人 100101812
弁理士 勝村 紘
- (74)代理人 100124394
弁理士 佐藤 立志
- (74)代理人 100112807
弁理士 岡田 貴志
- (74)代理人 100111073
弁理士 堀内 美保子
- (74)代理人 100134290
弁理士 竹内 将訓
- (74)代理人 100127144
弁理士 市原 卓三
- (74)代理人 100141933
弁理士 山下 元
- (72)発明者 松田 恭平
東京都港区芝浦一丁目1番1号 株式会社東芝内
- (72)発明者 国分 智幸
東京都港区芝浦一丁目1番1号 株式会社東芝内
- Fターム(参考) 5E501 AA03 BA02 BA03 CB05 DA16 EA11 FA43 FA46