

EXHIBIT 4.17

5

【0030】すなわち、入力デバイス識別部9は、入力座標検出部6で検出された座標データを用い、まず、今回の入力において、1つ以上のセンサが反応したか否かを判定する(ステップ100)。

【0031】つぎに、入力デバイス識別部9は、反応したセンサ数が10個~28個の範囲内に入っているか否かを調べ(ステップ110)、入っている場合は、関係式

$$X_{\min}+6 \leq X_{\max} \leq X_{\min}+4$$

$$y_{\min}+5 \leq y_{\max} \leq y_{\min}+7$$

が成立するか否かを調べ(ステップ120)、成立する場合に、今回の入力デバイスは細ペンであると認識する(ステップ130)。なお、 X_{\min} 、 y_{\min} は反応したセンサにおけるx値の最小値、 y_{\min} 、 y_{\max} は反応したセンサにおけるy値の最大値を示す。

【0032】つぎに、入力デバイス識別部9は、反応したセンサ数が80個~136個の範囲内に入っているか否かを調べ(ステップ140)、入っている場合は、関係式

$$X_{\min}+6 \leq X_{\max} \leq X_{\min}+8$$

$$y_{\min}+15 \leq y_{\max} \leq y_{\min}+17$$

が成立するか否かを調べ(ステップ150)、成立する場合に、今回の入力デバイスは太ペンであると認識する(ステップ160)。

【0033】つぎに、入力デバイス識別部9は、反応したセンサ数が60個~100個の範囲内に入っているか否かを調べ(ステップ170)、入っている場合は、関係式

$$X_{\min}+8 \leq X_{\max} \leq X_{\min}+10$$

$$y_{\min}+8 \leq y_{\max} \leq y_{\min}+10$$

が成立するか否かを調べ(ステップ180)、成立する場合に、今回の入力デバイスはイレーサ(小)であると認識する(ステップ190)。

【0034】つぎに、入力デバイス識別部9は、反応したセンサ数が400個~484個の範囲内に入っているか否かを調べ(ステップ200)、入っている場合は、関係式

$$X_{\min}+20 \leq X_{\max} \leq X_{\min}+22$$

$$y_{\min}+20 \leq y_{\max} \leq y_{\min}+22$$

が成立するか否かを調べ(ステップ210)、成立する場合に、今回の入力デバイスはイレーサ(大)であると認識する(ステップ220)。

【0035】また、上記何れの条件にも当てはまらない場合は、人間の手などが入力面に触れたとして、今回の入力を無視する(ステップ230)。

【0036】なお、入力デバイス識別部9において細ペンと識別された場合は、描画処理部10で細線による描画処理が行われ、また太ペンと識別された場合は太線による描画処理が行われ、また小イレーサと識別された際は小イレーサの軌跡に従った消去処理が行われ、また大

6

イレーサと識別された場合は大イレーサの軌跡に従った消去処理が行われる。

【0037】このように上記実施例では、入力デバイスのタッチ面積に応じて入力デバイスの種類を識別し、この識別結果に応じてその後の処理モード選択するようにしている。

【0038】なお、上記実施例において、入力デバイスの入力形状に応じて入力デバイスの種類を識別するようにしてもよい。この場合、入力デバイスの入力面の形状を各入力デバイス毎に変化させ、スクリーンパネルパネル側では入力された領域の形状を周知の図形認識技術を用いて認識させるようにすればよい。

【0039】また、上記実施例では、入力ペンとイレーサとは別体としたが、入力デバイスとして、図8に示すような、消しゴム付き入力ペンを用いることができる。すなわち、図8の入力デバイスにおいては、ペン側の芯部14で線描画を行い、ペン後端のイレーサ部分15でイレーサ処理を行うようにしており、通常の消しゴム付き鉛筆の感覚で手書き入力を行うことができる。

20 【0040】また、図2(b)に示した直方体形状のイレーサ5は、その形状および面積からみて、3つの異なる面を有している。したがって、これら3つの面の接触面積を入力ペンによる接触面積と異ならせるようにしておけば、このイレーサ5の各面を用いて、消去面積の異なる消去処理を行うことができる。

【0041】また、接触面積の異なる入力ペンを複数個用意し、これら複数個の入力ペンに対し、それぞれ異なる線属性(例えば、色、太さ、種類(実線、点線など))を設定するようにしておけば、入力ペンを選択することで線属性を変化させることもできる。例えば、線属性として色を設定すれば、色鉛筆を使う感覚で線描画の色を変化させることができる。

30 【0042】さらに、描画した線の属性として、消去可能なイレーサを選別するための消去レベルを設定し、この消去レベルを用いて消去可能なイレーサを選別するようにしてもよい。例えば、描画した線の属性として、図9のイレーサ20の面Aでのみ消去可能な消去可能レベル1と、イレーサの20のどの面でも消去可能な消去可能レベル2と、このイレーサ20では消去ができなく、別の形状の砂消しイレーサ(図示せず)でのみ消去可能な消去不可レベルを設定するようにすれば、砂消しごむと通常の消しゴムのような感覚で、油性ペン、ボールペン、鉛筆などに例えられるような属性を持つ描画に対する消去処理を行うことが可能になる。さらに、ブラシのような入力デバイスに修正液の機能を持たせ、上記イレーサ20では消去の不可能な描画に対して、上書きによる修正を可能にするようにしてもよい。

50 【0043】さらに、図10に示すように、コンパス機能を持つ入力デバイスを用意し、手書きスクリーンパネル1上で円を描画させるようにしてもよい。なお、この

場合、コンパスの支点21と描画点22を識別させるために、これらの面積および形状を異ならせている。

【0044】また、上記実施例では、感圧方式のパネルに本発明を適用するようにしたが、他の電磁授受方式、静電容量結合式、超音波式等の入力座標検出方式に本発明を適用するようにしてもよい。

【0045】図11は電磁授受方式による入力座標検出のための原理的構成を示すもので、入力ペンやイレーサなどの入力デバイス30には、コイル31およびコンデンサ32からなる共振回路を内蔵している。また、表示面としてのスクリーンパネルP（図示せず）の裏面には複数のX軸方向センサアンテナコイル33a、33b、33c、…をX軸方向に並設し、同様に複数のY軸方向センサアンテナコイル（図示せず）をY軸方向に並設する。

【0046】そして、まず送受信切替え用のスイッチ34の接点を交流電源36側に接続するとともに、コイル切替え用のスイッチ35をコイル33a、33b、33c、…を順次選択するよう高速で切替え、これにより各コイル33a、33b、33c、…に交流電源36からの交流電流を順次流し、各コイル33a、33b、33c、…によってスクリーンパネルP上に順次磁界を発生させる。

【0047】この結果、スクリーンパネルP上の任意の位置にある入力デバイス30の共振回路にスクリーンパネル上に発生した磁界に対応する電流が流れる。

【0048】この状態で、送受信切替え用のスイッチ34の接点を受信回路37側に切替えるとともに、各コイル33a、33b、33cを順次選択するようスイッチ35を高速で切り換えることにより、各コイル33a、33b、33cに発生される電流を受信回路37で順次検出する。

【0049】すなわち、入力デバイス30のコイル31から発生した磁界によって各コイル33a、33b、33cに発生した電流を受信回路37で検出し、この検出に基づいて入力デバイス30のX位置を特定する。すなわち、入力デバイス30の位置に最も近いX軸コイルに最も大きな電流が発生するので、各コイルに発生した電流を調べることによって入力デバイス30のX位置を特定することができる。

【0050】同様にして、入力デバイスのY位置も特定する。

【0051】このような電磁入力方式において、本発明を適用する場合は、図12に示すような入力デバイス30を用いる。すなわち、図12(a)は細書き用の入力ペンを示しており、コイル31およびコンデンサ32から成る共振回路が1個内蔵されている。図12(b)は太書き用の入力ペンを示しており、該ペン自体の太さ（径）は細書き用のペンの2倍であり、その中に計4（2×2）個の共振回路が内蔵されている。図12(c)は直方

体形状のイレーサを示し、その入力面の1辺の長さは細書き用ペンの直径の4倍であり、その内部に計16個（4×4）個の共振回路が分散配置されている。

【0052】このように、共振回路の個数が異なる複数の入力デバイス30を用いるようにすれば、例えば、図12(a)に示す細書き用の入力ペンはスクリーンパネルP上の1つのXY座標でその位置が検出され、また図12(b)に示す太書き用の入力ペンはスクリーンパネルP上の数個のXY座標でその位置が検出され、また図12(c)に示すイレーサはスクリーンパネルP上の10数個のXY座標でその位置が検出されるようになり、これに基づいて入力操作を行った入力デバイスの種類を識別することができる。

【0053】図13は、静電容量結合方式による入力座標検出のための原理的構成を示すもので、入力パネル40は、2枚のガラス上にX電極パターン及びY電極パターンをそれぞれ形成し、これら電極パターンが形成された2枚のガラスをその電極面が向かい合うように張り合わせるることによって形成される。かかる入力パネル40のY電極 $y_1 \sim y_n$ に、図14に示すような、時系列パルスを順次印加するとともに、X電極 $x_1 \sim x_n$ にも、同様の時系列パルスを順次印加する。この状態で、入力ペン（スタイラスペン）50を入力パネルに接近させれば、ペン先と電極間に生じる静電容量により、入力ペンが近づいた位置に対応する電極に印加されたパルスにはなまりが発生する。このなまりをX電極側およびY電極側のそれぞれに関して検出することにより、入力位置座標を特定する。

【0054】このような静電容量結合方式において、本発明を適用する場合は、図15に示すように、入力デバイス50の大きさ（径）を変化させる。すなわち、図15(a)は細書き用の入力ペンを示し、図12(b)は太書き用の入力ペンを示し、図12(c)はイレーサを示しているが、これら各入力デバイスの径は異なっている。

【0055】これにより、入力パネル40側においては、前記なまりの発生するXY座標の個数が入力デバイスに応じて変化する事になり、これに基づいて入力操作を行った入力デバイスの種類を識別することができる。

【0056】

【発明の効果】以上詳細に説明したようにこの発明によれば、手書き入力デバイスの入力面積または入力形状を検出し、この検出した入力面積または入力形状に基づいて入力操作を行った手書き入力デバイスの種類を識別し、該識別した種類に対応する描画処理を実行するようにしているので、操作者に特別な選択操作を行わずことなく手書き入力デバイスの種類の識別が可能となり、コストパフォーマンスおよび操作性が従来に比べ格段に向上する。また、文房具を用いる感覚で各入力デバイスを用いることができるので、幼児を対象とした手書き入力装置にも有効に適用できる。

【図面の簡単な説明】

- 【図1】この発明の実施例を示すブロック図。
- 【図2】入力デバイスを例示する図。
- 【図3】感圧パネルのXY電極の配置態様を示す図。
- 【図4】感圧パネルのY電極に対する印加パルスを示す図。
- 【図5】各種入力デバイスの入力形状を示す図。
- 【図6】入力デバイス識別処理の一例を示すフローチャート。
- 【図7】入力座標検出パネルの概念的構成を示す図。
- 【図8】他の入力デバイスを示す図。
- 【図9】この発明の変形例を説明するための図。
- 【図10】他の入力デバイスを示す図。
- 【図11】電磁授受方式による入力座標検出のための原理図。

*【図12】電磁授受方式に本発明を適用した場合の入力デバイスの構成を示す概念図。

【図13】静電容量結合方式の入力パネルを示す図。

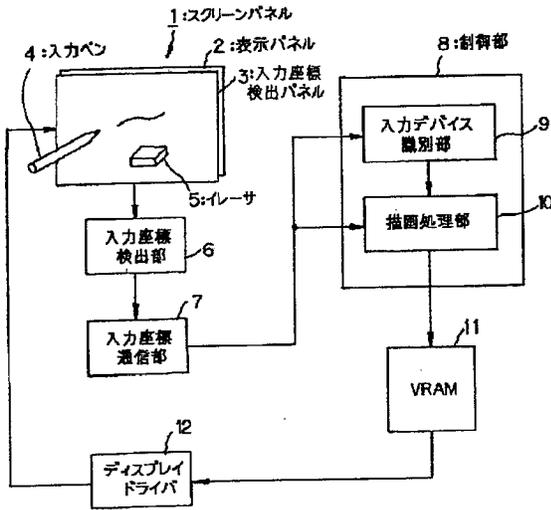
【図14】静電容量結合方式の入力パネルのY電極に対する印加パルスを示す図。

【図15】静電容量結合方式に本発明を適用した場合の入力デバイスの構成を示す概念図。

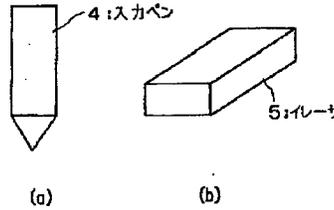
【符号の説明】

- 1…スクリーンパネル
- 2…表示パネル
- 3…入力座標検出パネル
- 4…入力ペン
- 5…イレーサ
- 6…入力座標検出部
- 7…入力座標通信部
- 8…制御部
- 9…入力デバイス識別部
- 10…描画処理部
- 11…VRAM (ビデオラム)
- 12…ディスプレイドライバ

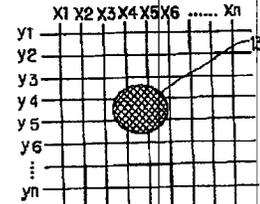
【図1】



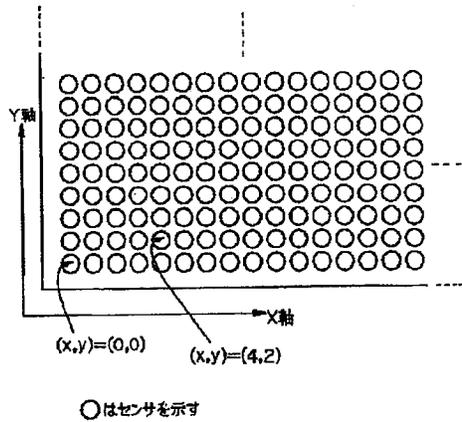
【図2】



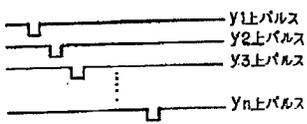
【図3】



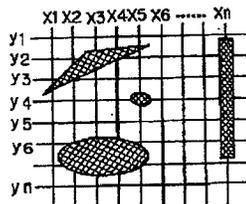
【図7】



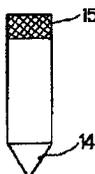
【図4】



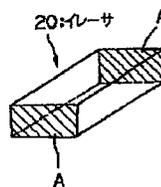
【図5】



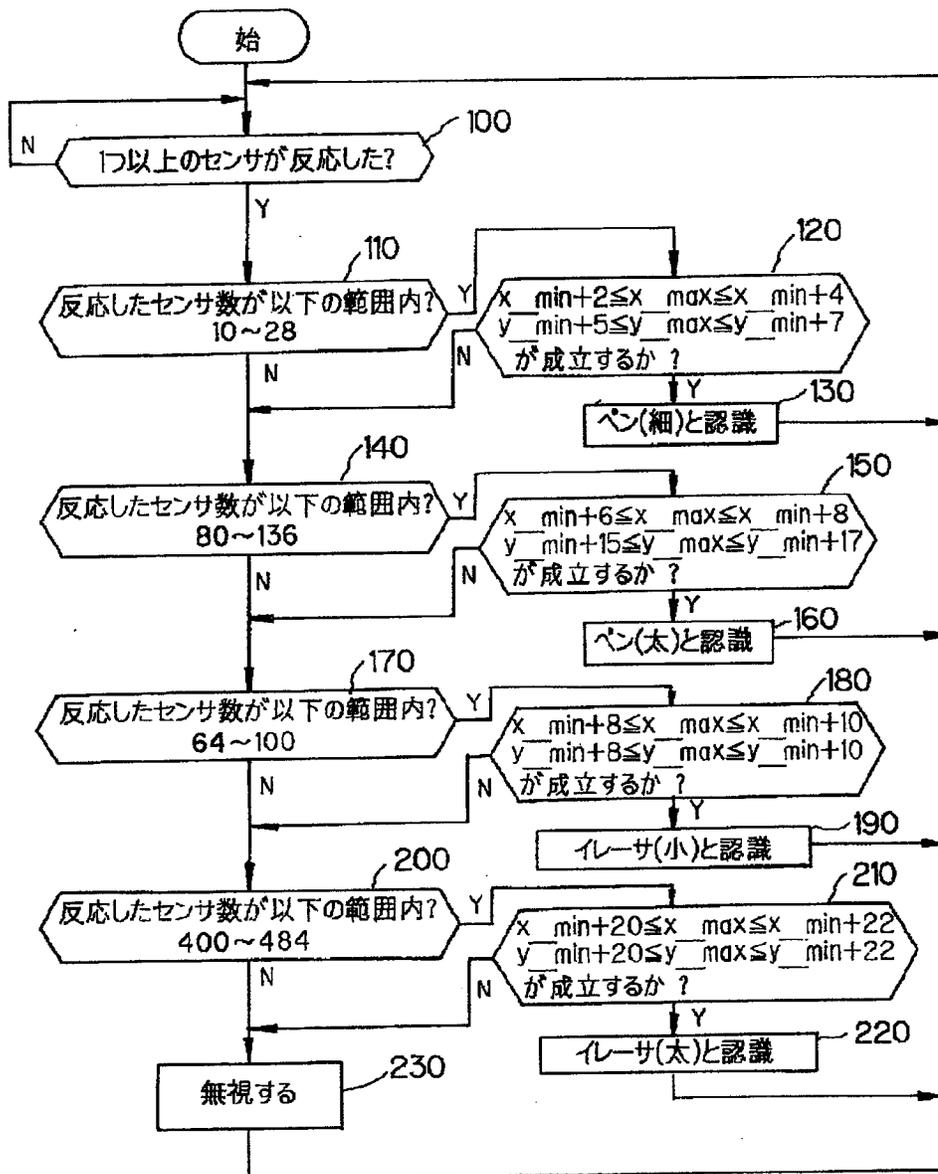
【図8】



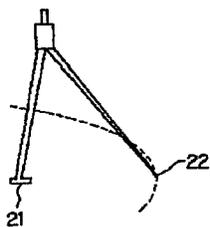
【図9】



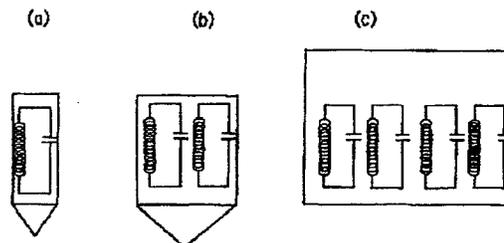
【図6】



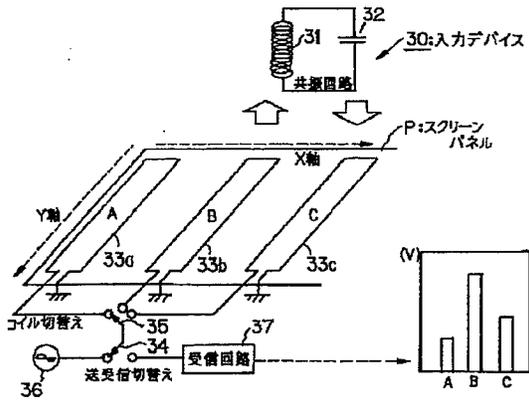
【図10】



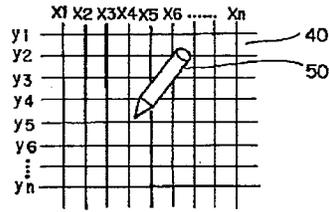
【図12】



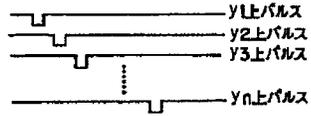
【図11】



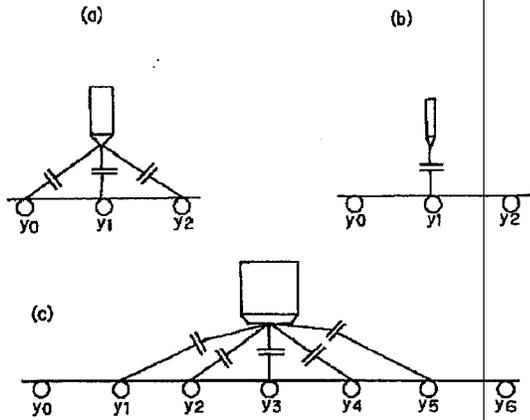
【図13】



【図14】



【図15】



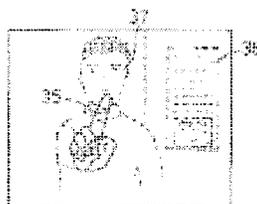
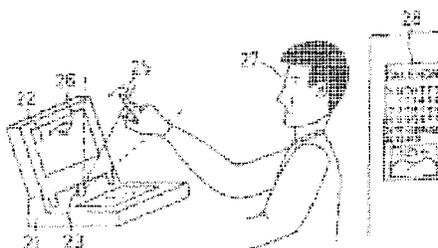
INFORMATION PROCESSOR

Patent number: JP7057103 (A)
Publication date: 1995-03-03
Inventor(s): IDE YUJI; SASUGA MITSUO; TAKASHIMA KAZUHIRO; MOROHOSHI TOSHIHIRO; NAKAJIMA KOJI +
Applicant(s): TOSHIBA CORP +
Classification:
 - international: G06F3/033; G06F3/038; G06T1/00; G06T7/20; G06T9/20; G06F3/033; G06T 1/00; G06T7/20; G06T9/20; (IPC1-7): G06T9/20; G06F3/033; G06T1/00
 - european:
Application number: JP199302073 58 19930823
Priority number(s): JP199302073 58 19930823

Abstract of JP 7057103 (A)

PURPOSE:To place the information processor in operation by operator's operation at a specific position by picking up an image of the specific position of the operator by an image pickup means, recognizing the shape or position of the contour, and performing control by using the result.

CONSTITUTION:When the operator 27 points his or her index finger 25 at a display screen 22, a cursor 26 is displayed on the display screen 22 corresponding to the position of the tip of the index finger 25. An image pickup lens 23 has shallow depth of field; and only a range of about 5-25cm from the front surface of the lens is put in focus and a subject which is a >25cm away becomes out of focus. Namely, an image of the hand including the index finger 25 is picked up in the in-focus state and a background image 28 of the head 27 of the operator 27 behind it and others is picked up in an out-of-focus state. The part of the hand 35 of the operator is in focus, so an image signal of the hand includes a high-frequency component and the contour is sharp, but an image signal of the face 27 and background 38 includes no wide-range frequency component, so the contour is not sharp.



Data supplied from the *espacenet* database — Worldwide

(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

特開平7-57103

(43)公開日 平成7年(1995)3月3日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 T 9/20				
G 0 6 F 3/033	3 1 0 Y	7165-5B		
G 0 6 T 1/00				
		8837-5L	G 0 6 F 15/ 70	3 3 5 Z
		9287-5L	15/ 62	3 8 0
			審査請求 未請求	請求項の数4 OL (全 7 頁)

(21)出願番号 特願平5-207358

(22)出願日 平成5年(1993)8月23日

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 井手 祐二

神奈川県川崎市幸区小向東芝町1番地 株式会社東芝研究開発センター内

(72)発明者 流石 三夫

神奈川県川崎市幸区小向東芝町1番地 株式会社東芝研究開発センター内

(72)発明者 高島 和宏

神奈川県川崎市幸区小向東芝町1番地 株式会社東芝研究開発センター内

(74)代理人 弁理士 則近 憲佑

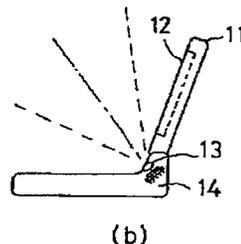
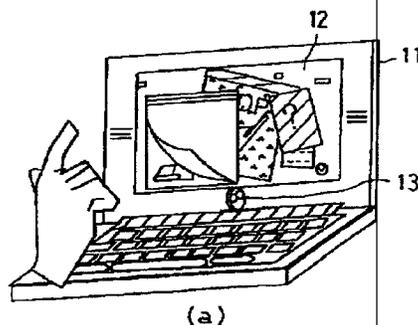
最終頁に続く

(54)【発明の名称】 情報処理装置

(57)【要約】

【目的】 本発明の目的とするところは、マウス等の特殊なインターフェイス装置なしで操作者の指等の特定部位を空間上で動かすだけでポインタ操作が可能となり、あるいは動作を認識し、コンピュータやマルチメディア機器に対し、起居動作を用いたコマンド操作を可能にするコンピュータシステムを提供することにある。

【構成】 情報データを処理する情報データ処理部と情報データに関係する表示を行う表示部を備えた情報処理装置において、情報処理装置を操作する操作者の特定部位を撮像する撮像手段と、この撮像手段で撮像された特定部位の輪郭を抽出する輪郭抽出手段と、この輪郭抽出手段で抽出された特定部位の輪郭の形状または位置を認識する特定部位認識手段と、この特定部位認識手段の認識結果に基づいて情報データ処理部が表示部の内少なくとも一方を制御する制御手段を具備したことを特徴とする。



【特許請求の範囲】

【請求項1】 情報データを処理する情報データ処理部と、前記情報データに関係する表示を行う表示部を備えた情報処理装置において、

前記情報処理装置を操作する操作者の特定部位を撮像する撮像手段と、

この撮像手段で撮像された前記特定部位の輪郭を抽出する輪郭抽出手段と、

この輪郭抽出手段で抽出された前記特定部位の輪郭の形状または位置を認識する特定部位認識手段と、

この特定部位認識手段の認識結果に基づいて、前記情報データ処理部か前記表示部の内少なくとも一方を制御する制御手段とを具備したことを特徴とする情報処理装置。

【請求項2】 撮像手段は、前記表示部の下方部ではほぼ中央の位置に設けられ、撮像光軸をほぼ斜め上方に傾けてなることを特徴とする請求項1記載の情報処理装置。

【請求項3】 特定部位認識手段は、前記特定部位の輪郭の中で少なくとも一つの先端部を検出し、この先端部の動きを認識することを特徴とする請求項1記載の情報処理装置。

【請求項4】 情報データを処理する情報データ処理部と、前記情報データに関係する表示を行う表示部を備えた情報処理装置において、

前記情報処理装置を操作する操作者の特定部位を撮像する撮像手段と、

この撮像手段で撮像された前記特定部位の輪郭を抽出する輪郭抽出手段と、

この輪郭抽出手段で抽出された前記特定部位の輪郭の形状または位置を認識する特定部位認識手段と、

予め前記特定部位の輪郭の形状または位置に基づく複数種類のパターンに対応づけて、情報データ処理部か表示部の内少なくとも一方を制御する制御情報を記憶する記憶手段と、

前記特定部位認識手段の認識結果に応じたパターンに対応する前記制御情報を前記記憶手段から読み出し、この読み出された制御情報に基づいて、前記情報データ処理部や前記表示部を制御する制御手段を具備したことを特徴とする情報処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、操作者の指動作で情報の制御を行うパーソナルコンピュータや携帯型情報機器やマルチメディア機器等の情報処理装置に関する。

【0002】

【従来の技術】従来、コンピュータ等へのコマンド入力や制御にはキーボードによる文字入力やマウスと呼ばれるポインタ装置、またはペン入力によるポインタ操作や文字入力操作が広く用いられている。

【0003】キーボードによる操作は熟練を要するが、

文字列の入力には速度と正確さの点で現在でも主流のマン・マシン・インターフェイスになっている。しかし、より操作しやすいインターフェイスとして、画面上の任意の点をポインティングできるマウスが近年急速に普及してきた。また、キーボードを搭載できない携帯型情報機器やキーボード操作に不慣れなユーザーでも使用できるペン入力コンピュータが出現してきた。

【0004】以下マウスの一例とペン入力コンピュータの一例について図を用いて説明する。マウスにはメカニカル式マウスと光学式マウスに大別できるが、メカニカル式マウスの方が特殊な操作板を必要としないためより普及している。

【0005】図13にメカニカル式マウスの概略構成図を示す。メカニカル式マウスの代表的なものは回転球を用いるもので、図13に示すようにマウス本体130の下部に突出した球131が机等の操作面132の上で回転可能なように取り付けられている。玉の回転を縦、横2方向に検出する回転検出機構133、134を回路基板135上に設け、動きを電気的な信号に変換して本体の動きを検出している。操作者は平な操作面132の上をマウス本体130を動かしながら、図示しない表示画面上のカーソルを動かし、クリックボタン136を押してクリック動作をする。マウスの制御信号はケーブル137を介して伝送される。マウス本体130は、コネクタ138を用いてコンピュータ等の制御対象の情報処理装置へ接続される。このメカニカル式では平面上の動きしか捕らえられず、またマウス操作のためにある程度(25cm×25cm)の大きな平らな操作面が必要であるという欠点がある。

【0006】このようなマウスを用いた操作では、表示画面のポインティング動作とクリック動作しかできず、また表示画面のウィンドウをクリックする時ウィンドウ内の小さな場所にカーソルを合わせてクリックするために不便であり、操作装置を介して行うため日常生活の動作で行うような感覚的操作とは言えない。

【0007】図14はペン入力コンピュータの概略構成図を示す。ペン入力コンピュータ本体141のカバー142を開くと大きな液晶表示パネル143がある。操作者がこの表示パネル143上の任意の位置を専用ペン144の先で触れるとカーソルが表示されポインティング動作を行うことができる。また、表示パネル143上にペン144で手書きの文字を書くとコンピュータ側で文字認識を行い、内蔵するキャラクタ発生装置から発生された文字へ自動変換し、きれいな文字へ変換表示することができる。あるいは、手書きの地図や絵もグラフィック情報として記憶することができる。

【0008】しかし、あくまで1本のペンで表現できる範囲の情報やコマンドしか入力できず、また操作途中でペンを持ったり置いたりする操作がめんどうで、さらに携帯操作ではペンを落としたりすることもあったため、

ヒューマン・インターフェイスとしては表現力と操作性に問題があった。

【0009】

【発明が解決しようとする課題】以上述べてきた様に従来のマウスを用いたコンピュータ等では、平な作業台や専用の操作板が必要で空間上で操作をすることができなかつた。また操作機器を利用する為操作動作が感覚的に行えず不便な情報処理装置であった。

【0010】あるいはペン入力機能を有するコンピュータ等でも、1本のペンで情報処理装置の入力画面の平面上で表現可能な情報やコマンドしか入力できず、またペンを持ちたり置いたりめんどうで携帯操作時にペンを落としたりすることがあるため、表現力と操作性に大きな問題があった。

【0011】本発明はこのような問題を解決するためになされたもので、その目的とるところは、特殊なインターフェイス装置なしで空間上で操作者の指等の特定部位を用いてポインタ操作を可能とし、あるいは指等の操作者の特定部の動作を認識しコンピュータやマルチメディア機器に対し、起居動作を用いたコマンド操作を可能にする情報処理装置を提供することにある。

【0012】

【課題を解決するための手段】上記目的を達成するために本発明においては、情報データを処理する情報データ処理部と、情報データに関係する表示を行う表示部を備えた情報処理装置において、情報処理装置を操作する操作者の特定部位を撮像する撮像手段と、この撮像手段で撮像された特定部位の輪郭を抽出する輪郭抽出手段と、この輪郭抽出手段で抽出された特定部位の輪郭の形状または位置を認識する特定部位認識手段と、この特定部位認識手段の認識結果に基づいて、情報データ処理部が表示部の内少なくとも一方を制御する制御手段とを具備したことを特徴とするものである。

【0013】

【作用】操作者の特定部位を撮像手段で撮像し、特定部位認識手段による特定部位の輪郭の形状または位置を認識して、その結果に基づいて制御手段が情報データ処理部や表示部の制御をしているので、操作者の特定部位の動作で情報処理装置を操作することが可能となる。

【0014】

【実施例】以下、本発明の一実施例を図面を参照して説明する。図1は本発明の一実施例に関わる指先動作制御型コンピュータの外観図(a)と側面図(b)を示す。コンピュータ本体11に取り付けられた表示パネル12の下方ではば左右中央の位置に、被写界深度が浅く、かつ撮像光軸を斜め上方に傾けた撮像レンズ13が設けられている。撮像レンズの後方には、固体撮像素子14が設けられており、表示パネル12の付近傍付近の映像を撮像することができる。ここで、撮像レンズ13が表示パネル12の下方に取付られているのは、手を下方から撮像する

ことで指先の先端が常に手の他の部分より上に位置する角度で撮像できるようにするためである。

【0015】図2に本発明の指先動作制御型コンピュータを操作している状態の斜視図を示す。操作者27が表示画面22の前で人指し指25を指し示すと、人指し指25の先端の位置に対応して表示画面22上にカーソル26が表示される。撮像レンズ23は被写界深度が浅くレンズの全面から約5cm~25cmの範囲だけピントが合い、25cm以上離れた被写体はピントがボケてしまう。すなわち、人指し指25を含む手の画像にはピントが合い、後方の操作者の顔27やその他の背景画像28はボケて撮像されている。このような撮像画像の一例を図3に示す。操作者の手35の部分はピントが合っているため、この手35の画像信号には高域周波数成分が含まれており輪郭がはっきりしているが、他の顔37や背景38の画像信号には高域周波数成分が含まれていないため輪郭がはっきりしていない。

【0016】図4に指先動作制御型コンピュータの指先位置信号処理部の概略ブロック図を示す。撮像レンズ41を通して入射した手の映像は撮像素子42上に結像され、その画像出力信号から輪郭抽出回路43で高域周波数成分が抽出され先端検出回路44に入力される。先端検出回路44からの指先位置信号は表示位置データ変換回路45に入力され、ビデオ信号記憶用ランダムアクセスメモリ(VRAM)46に入力された後、表示画面である液晶表示パネル(LCD)47上に表示される。

【0017】これらの信号処理を図3~図5を用いてさらに詳しく説明する。図5は輪郭抽出回路43で得られた撮像画像である。図3に示した通り、撮像レンズ41は被写界深度が浅く、レンズ全面の5cmから25cm程度の範囲の対象物しかピントが合っていないために、輪郭がはっきりとしているのは操作者の手35しかない。そのために、図4に示した通り、撮像素子42から得られる映像の輪郭信号を抽出すると図5に示したような手の輪郭信号52のみを抽出することができる。この輪郭信号を先端検出回路44に入力し、入力画像信号51の走査線の上から順番に輪郭信号の有無を判定していく。例えば、1番目の走査線53まではまだ輪郭信号は何もなく、1+m番目の走査線54ではじめて輪郭信号が検出される。この輪郭信号が検出された位置をP点とすると、P点の水平方向の画素番号nと垂直方向の画素番号1+mの位置を、表示位置データ変換回路45により数値データ化して、メモリ46にカーソル形状として一時記憶させ、図示しないコンピュータからの他の表示信号と同時に表示部47上に表示する。図6はその表示部47上の表示画面の一例である。コンピュータ本体61の表示部としてLCD62があり、その下部ではば左右中央の位置に撮像レンズ67がある。LCD62の表示画面にはP点に相当する位置63、則ち水平n番画素、垂直1+m番画素の位置にカーソル64が表示され

ている。但しこの場合P点の水平位置は、撮像画像と表示画像で左右が反転するために、右からn番目の位置である。この表示画面62にはコンピュータ61からの他の画像65、66も同様に表示されている。

【0018】次に図7と図8を用いて本発明の第2の実施例について述べる。図7は本発明の第2の実施例の指先動作制御型コンピュータの指先動作信号処理回路の概略ブロック図を示す。第1の実施例と同様に、撮像レンズ71を通して入射した手の映像是撮像素子72上に結像され、その画像出力信号から輪郭抽出回路73で高域周波数成分が抽出され先端検出回路74に入力される。先端検出回路74からの指先位置信号は表示位置データ変換回路75に入力され、ビデオ信号記憶用ランダムアクセス用のメモリ76に一時記憶された後、表示画面である液晶表示パネル等の表示部77上に表示される。ここまでは第1の実施例と同様であるが、指先動作認識回路78で輪郭抽出回路73からの輪郭信号と先端検出回路74からの指先端位置信号をもとに指先の相対的動作を認識し、この動作認識結果に基づいてコマンド発生回路79からコンピュータに対するコマンドが送出される。

【0019】図8はこの第2の実施例に関わる操作者の右手の輪郭信号81を示した図である。この時、操作者は人指し指82と親指83を伸ばしている。第1の実施例と同様に画面上方から順番に先端を2つ検出し、左側を人指し指の先端84、右側を親指の先端85とし、2つの先端の動作を認識する。例えば、人指し指と親指の先端の2点84、85が相対的に逆方向の運動をしたら回転運動と判定し、カメラ映像で左回転の場合は操作者が右回転の指動作をしたものと理解し、表示画面内のページまたはウインドウをめくったり、表示画面内のオブジェクトを回転させたりする。逆に、カメラ映像で右回転の場合は操作者が左回転の指動作をしたものと理解する。さらに、人指し指と親指の先端の2点84、85が近接したら、摘み動作と理解し表示画面内のオブジェクトを摘み、摘んだまま移動することができる。あるいは、人指し指と親指の先端の2点84、85の距離に比例して表示真画面内のオブジェクトまたはページあるいはウインドウの大きさを変化させ、拡大、縮小させることができる。また、任意の指の相対的大きさを輪郭信号より検出し、その大きさに応じて、表示画面内のオブジェクトの大きさを変化させたり、奥行き方向の表現制御をしたりする3次元制御ができる。このように、操作者の意志を指の形状や動作から認識し、コンピュータへコマンドとして指示することができる。

【0020】さらに、本発明の第3の実施例について図9を用いて説明する。図9(a)は操作者の右手の人指し指と中指を用いて、丸を作り「OK:承認」の意味を示したときの輪郭信号を示した図である。又、図9

(b)は操作者の右手の親指を立てて「check:確

認」あるいは「good:良好」の意味を示したときの輪郭信号を示した図である。図9(c)は人指し指を立てて横に振り、「no:否認」の意味を示した時の輪郭信号である。図9(d)は人指し指と中指を立てて「鉄」の意味を示した時の輪郭信号で、表示画面内の任意の場所を切り抜く時に用いる。これらの基準となる手話の手形状の輪郭信号パターンを手話パターン記憶手段に予め登録し、その登録手話パターンと撮像画像の輪郭信号から得られる手話パターンとの比較を行い、操作者の意志を手話にて理解しコンピュータのコマンドとして伝えることができる。

【0021】図10(a)~(d)に図9(a)~(d)の手話に対応した基準となる登録手話パターンを示す。輪郭信号の内でも境界をはっきりしている手話形状の外形線のみで登録してある。図11に手話を用いる指先制御型コンピュータの手話パターン信号処理部の概略ブロック図を示す。撮像レンズ111を通して入射した手の映像是固体撮像素子112上に結像され、その画像出力信号から輪郭抽出回路113で高域周波数成分が抽出され手話パターン検出回路114に入力される。手話パターン検出回路114では、輪郭信号を基に手話形状の外形線が検出され、比較・判断回路114に操作者の手話パターンとして入力される。比較・判断回路115に操作者の手話パターンとして入力される。比較・判断回路115では、基準の登録手話パターンが記憶されている登録手話パターン記憶回路116の記憶パターンと逐次比較を行い、その結果を基にコマンド発生回路117からコンピュータに対するコマンドが送出される。

【0022】これらの手話動作はいずれも日常生活で用いている一般的な動作で、いわば体で覚えている動作である。このような動作を簡単な手話として用いコンピュータやマルチメディア機器の制御を行うと、特別な訓練なしにコマンド操作が可能になる。

【0023】次に図12を用いて本発明の第4の実施例について説明する。本発明の第1~第3の実施例では、コンピュータに内蔵された撮像装置により操作者の手を対象に撮像していたが、このようなコンピュータシステムでは、1個の撮像装置を有効に用いる目的でTV会議や画像電子メール用の顔画像入力装置として用いたいと言う要求が出てくる。しかし、本発明の第1~第3の実施例で述べたような被写界深度が浅い撮像レンズでは操作者の顔を鮮明に撮像することはできない。そこで、第4の実施例ではコンピュータ本体に内蔵され撮像レンズの焦点距離を切り換える2焦点式撮像装置を設け、指動作は近距離撮像レンズで撮像し、顔は遠距離撮像レンズで撮像する。図12の例ではコンピュータ本体121に内蔵された近距離撮像レンズ(凸レンズ)122を用いて指動作を撮像する。操作者の顔124を撮像したいときには遠距離撮像用補正レンズ(凹レンズ)123を近距離撮像レンズ122の前に挿入してレンズ前面50c

m~1mの距離にある対象物にピントが合うようにすると、操作者の顔にピントが合い鮮明な顔画像125をコンピュータ121に入力することができる。

【0024】撮像レンズの焦点距離はコンピュータのコマンドあるいは手動切り換えて操作者が必要に応じてレンズを切り換えても良いし、自動で撮像レンズの全面5cm~25cmの範囲に対象物が何もないときには、近距離撮像モードから遠距離撮像モードに自動で切り換えても良い。また、焦点距離の切り換えは焦点距離補正レンズの挿抜、あるいは別途設ける焦点調節用レンズの位置を変化させても良いし、撮像レンズと撮像素子の距離を変化させても良い。さらに、レンズ絞りを変化させ被写界深度を調整しても良い。

【0025】尚、本発明の第1~第3の実施例で説明した撮像素子以降の各部の信号処理手段は、ハードウェアで行ってもソフトウェアで行っても信号処理としては本質的に何ら変わりがなく、本発明の主旨を逸脱しない範囲でハードウェアとソフトウェアによる信号処理手段を混合して構成できることは言うまでもない。

【0026】

【発明の効果】以上詳述したように本発明によれば、マウス等の外部の特殊なインターフェイス装置なしで操作者の指等の特定部位を空間上で動作させるだけでポインタ操作が可能となり、あるいは動作を認識し、コンピュータやマルチメディア機器に対し感覚動作でコマンド操作ができる。

【図面の簡単な説明】

- 【図1】 本発明の一実施例を示した図。
- 【図2】 本発明の制御動作を示した概念図。
- 【図3】 本発明の撮像手段で撮像された画像を示した図。
- 【図4】 本発明の一実施例を示したブロック図。

*【図5】 本発明の輪郭抽出部で得られる画像を示した図。

【図6】 本発明の表示部の一表示図面を示した図。

【図7】 本発明の第2の実施例を示した図。

【図8】 本発明の第2の実施例を示した図。

【図9】 本発明の第3の実施例を示した図。

【図10】 本発明の第3の実施例を示した図。

【図11】 本発明の第3の実施例を示したブロック図。

10 【図12】 本発明の撮像手段における他の実施例を示した図。

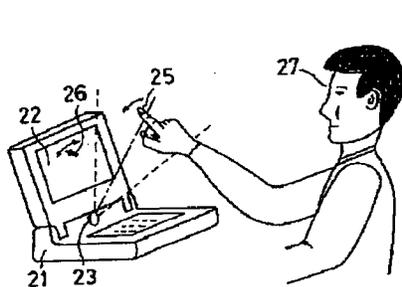
【図13】 従来例を示した図。

【図14】 従来例を示した図。

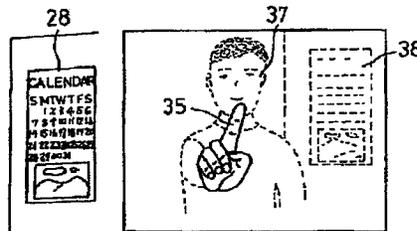
【符号の説明】

- 11、21、61、121…コンピュータ本体
- 12、22、62…表示部
- 13、23、41、67、111、122…撮像レンズ
- 14、42、112…撮像素子
- 25…指
- 27、124…操作者
- 43、113…輪郭抽出回路
- 44…先端検出回路
- 45…表示位置データ変換回路
- 47…液晶表示パネル
- 52、81…手の輪郭信号
- 64…カーソル
- 55、84、85…指の先端
- 123…遠距離撮像用補正レンズ
- 114…手話パターン検出回路
- 115…比較判断回路
- 116…登録手話パターン記憶回路

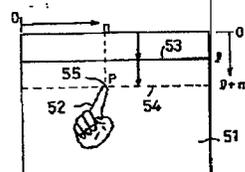
【図2】



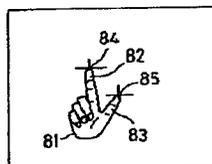
【図3】



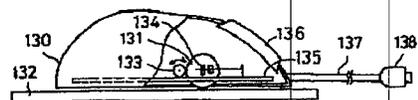
【図5】



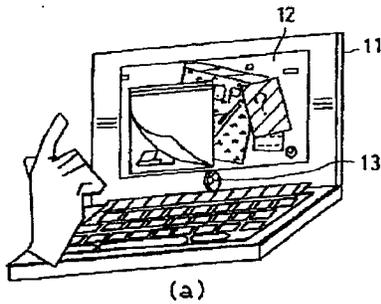
【図8】



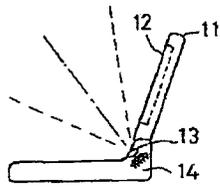
【図13】



【図1】

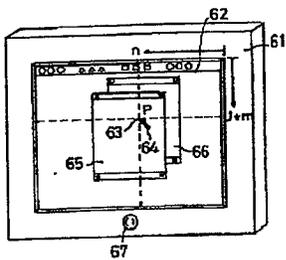


(a)



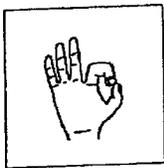
(b)

【図6】

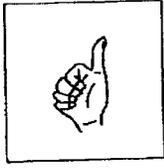


67

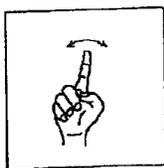
【図9】



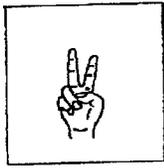
(a)



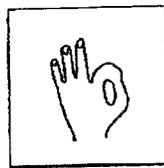
(b)



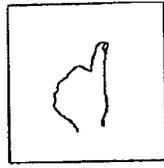
(c)



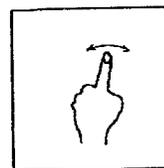
(d)



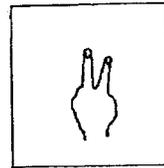
(a)



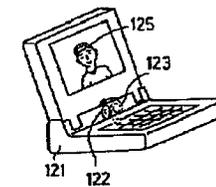
(b)



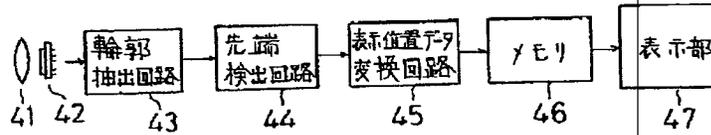
(c)



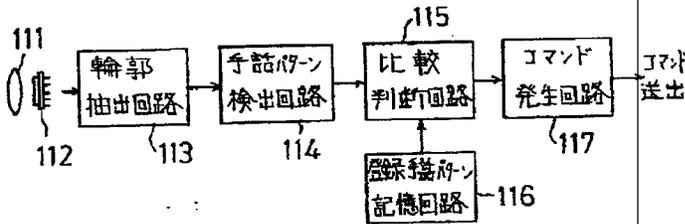
(d)



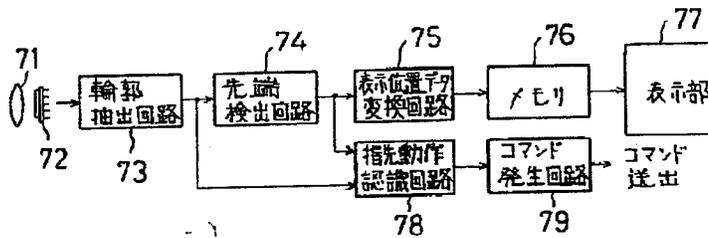
【図4】



【図11】



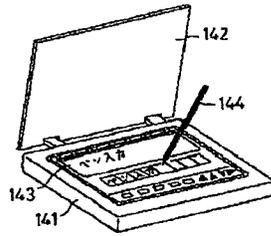
【図7】



【図10】

【図12】

【図14】



フロントページの続き

(72)発明者 諸星 利弘
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 中島 孝次
東京都港区芝浦一丁目1番1号 株式会社
東芝本社事務所内

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平6-332620

(43) 公開日 平成6年(1994)12月2日

(51) Int.Cl. ⁵	識別記号	弁内整理番号	F I	技術表示箇所
G 0 6 F 3/033	3 8 0 R	7 1 6 5 - 5 B		
3/03	3 1 5 B	7 1 6 5 - 5 B		

審査請求 未請求 請求項の数 3 F D (全 13 頁)

(21) 出願番号 特願平5-146931

(71) 出願人 000005267

プラザ工業株式会社

愛知県名古屋市瑞穂区苗代町15番1号

(22) 出願日 平成5年(1993)5月25日

(72) 発明者 加古 光政

名古屋市瑞穂区苗代町15番1号

プラザ

工業株式会社内

(72) 発明者 山本 好次郎

名古屋市瑞穂区苗代町15番1号

プラザ

工業株式会社内

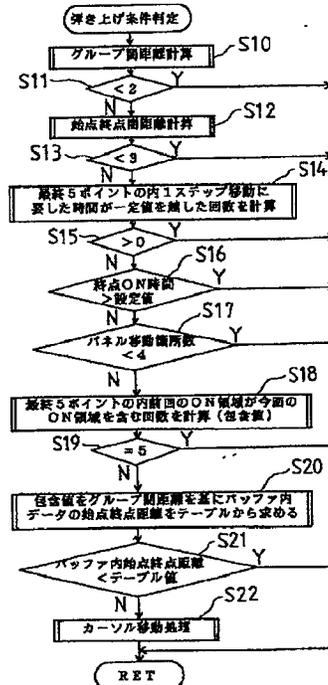
(74) 代理人 弁理士 山中 郁生 (外2名)

(54) 【発明の名称】 カーソル移動制御装置

(57) 【要約】

【目的】 パネルスイッチ上で弾き入力動作が行なわれた場合にはカーソルをディスプレイ上で迅速、且つ、飛躍的に移動させることができ、もってカーソル移動の操作性を向上することができるカーソル移動制御装置を提供する。

【構成】 パネルスイッチ上において指等により押下操作された場合、その押下操作が通常押下動作により行なわれたものであるか、又は、弾き上げ動作により行なわれたものであるかを判断し (S10乃至S22)、弾き上げ動作を伴って押下操作が行なわれたと判断された場合には終点グループにおける各ポイントのX方向及びY方向の平均移動速度に基づいて各X方向タイマ、Y方向タイマにセットされるタイマセット時間を漸増させつつ割り込み処理を行なう (S40乃至S54) とともに、タイマセット時間が所定の設定値以下になった場合にCRTディスプレイ上でカーソルを停止させるように構成する。



【特許請求の範囲】

【請求項1】 複数の導体ラインを形成した一対のフィルム基板と、各フィルム基板間に配置されるスペーサとを有し、各フィルム基板における各導体ラインがスペーサを介して相互に離間対向してマトリクス状に配設されてなるパネルスイッチと、カーソルを表示可能なディスプレイと、パネルスイッチ上で各導体ラインの接触位置が押下開始位置から押下終了位置まで移動されることに対応してディスプレイ上に表示されたカーソルを移動させるカーソル移動手段とを有するカーソル移動制御装置において、

前記押下終了位置で前記各導体ラインの接触が保持されている時間が所定時間以下であるかどうかを判断する第1判断手段と、

前記各導体ラインの接触位置が前記押下終了位置の近傍で移動する際に接触位置の移動速度が所定速度以上であるかどうかを判断する第2判断手段と、

前記第1判断手段により前記各導体ラインの接触が保持されている時間が所定時間以下であると判断され、かつ、前記第2判断手段により前記接触位置の移動速度が所定速度以上であると判断された場合、前記ディスプレイ上でカーソルを所定距離だけ移動させるに要する時間を算出する時間算出手段と、

前記カーソルを所定距離だけ移動させる毎に前記時間算出手段により算出された時間を一定の割合で増加させる時間増加手段と、を備え、

前記カーソル移動手段は、前記時間増加手段に増加された時間にて前記カーソルを所定距離だけ移動させることを特徴とするカーソル移動制御装置。

【請求項2】 前記時間増加手段により増加された時間が所定時間以上になった場合に、前記カーソルをディスプレイ上で停止させる停止手段をさらに備えたことを特徴とするカーソル移動制御装置。

【請求項3】 各種データを入力するキーボードを有し、前記パネルスイッチはキーボードに配設されていることを特徴とする請求項1記載のカーソル移動制御装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、複数のスイッチ部が配列されたパネルスイッチ上で指等を介して各スイッチ部のオン位置を移動させることに対応して、CRT等のディスプレイ上に表示されたカーソルの移動制御を行なうカーソル移動制御装置に関し、特に、操作者がパネルスイッチ上で指等を移動する際に、パネルスイッチ上で指等をスライドさせる通常の入力動作と指等を弾き上げる弾き上げ入力動作とを判別し、弾き上げ入力動作が行なわれた場合にはカーソルをディスプレイ上で迅速、且つ、飛躍的に移動させるようにしたカーソル移動制御装置に関するものである。

【0002】

【従来の技術】従来より、CRT等のディスプレイ上に表示されたカーソルの移動制御を行なうについて各種のカーソル移動制御方式が採用されている。かかる従来のカーソル移動制御方式では、例えば、「左」、「右」、「上」、「下」等のカーソルコントロールキーを操作することによりディスプレイ上のカーソルを移動させる方式、透明タッチパネルをディスプレイの前面に配設し入力ペンを用いてカーソルの表示位置を指定することによりカーソルの移動制御を行なう方式、ジョイスティックやライトペンによりカーソルの表示位置を指定してカーソルを移動させる方式等が一般に採用されている。

【0003】しかしながら、カーソルコントロールキーによりカーソルの移動制御を行なう方式では、多くの場合、一回のキー操作でカーソルを所望の位置まで移動させることはできず、何回もキー操作を繰り返す必要があり、また、カーソルが曲折した動きをすることから操作者の操作感覚に合致せず、これより作業能率が低下してしまい操作性が悪いという欠点があった。また、透明タッチパネルを使用してカーソルの移動を制御する方式では、キーボード等の入力装置に加えて更に透明タッチパネルをディスプレイに装備する必要があることから構成が複雑化するとともに故障率も高くなり、また、コスト高を招来するという欠点があった。更に、ジョイスティックやライトペンを介してカーソルの移動制御を行なう方式にあっては、構成が大がかりとなり、且つ、回路構成も複雑化し、故障率も高いという欠点があった。

【0004】このような各種制御方式における欠点を解決するため特公平1-46888号公報において、格子状に複数の導体群が配設されたタブレット上の所望の位置に入力ペンを当接させて所望の部位を指示するとともに、その指示された当接部位に係る座標データをタブレット制御回路から出力し、かかるタブレット制御回路から出力される座標データに関して相前後する当接部位の座標データの差の量だけCRT装置上にてカーソルを移動させるようにしたカーソル移動制御方式が提案されている。かかるカーソル制御方式によれば、操作者による入力ペンの動きに対応してCRT上に表示されたカーソルを追尾移動させて、操作者の操作感覚に合致させながらカーソルの移動制御を行なうことができ、作業能率の改善、操作性の向上等を図ることができるものである。

【0005】

【発明が解決しようとする課題】しかしながら、前記した特公平1-46888号公報に記載されたカーソル移動制御方式では、タブレット上の各部位をCRT上のカーソル表示箇所に対応させ、タブレット上の所望の部位を当接した場合には、その当接部位の座標データをタブレット制御回路から出力するとともに、その出力された座標データに関し相前後する当接部位の座標データの差の量に基づいてカーソルをCRT上で移動させるもので

ある。これより、このカーソル移動制御方式においては、大きなタブレットが必要となり、また、タブレットは文字等を入力することを主要目的としていることから、各座標データの差以上にカーソルを移動させる技術は全く存在していない。従って、タブレットを操作する際、操作者はかなり大きな動きをしながら操作することを強いられ、長時間の操作を行なうと疲労し易いという問題があり、また、迅速、且つ、飛躍的にカーソルを移動させることができないという問題も残存していた。

【0006】本発明は前記従来技術の問題点を解消するためになされたものであり、小型化を図ることによりキーボード等の入力装置に配設可能とするとともに、操作者がパネルスイッチ上で指等を移動する際に、パネルスイッチ上で指等をスライドさせる通常の入力動作と指等を弾き上げる弾き上げ入力動作とを判別し、弾き上げ入力動作が行なわれた場合にはカーソルをディスプレイ上で迅速、且つ、飛躍的に移動させることができ、もってカーソル移動の操作性を向上することができるカーソル移動制御装置を提供することを目的とする。

【0007】

【課題を解決するための手段】前記目的を達成するため本発明は、複数の導体ラインを形成した一対のフィルム基板と、各フィルム基板間に介挿されるスペーサとを有し、各フィルム基板における各導体ラインがスペーサを介して相互に離間対向してマトリックス状に配設されるパネルスイッチと、カーソルを表示可能なディスプレイと、パネルスイッチ上で各導体ラインの接触位置が押下開始位置から押下終了位置まで移動されることに対応してディスプレイ上に表示されたカーソルを移動させるカーソル移動手段とを有するカーソル移動制御装置において、前記押下終了位置で前記各導体ラインの接触が保持されている時間が所定時間以下であるかどうかを判断する第1判断手段と、前記各導体ラインの接触位置が前記押下終了位置の近傍で移動する際に接触位置の移動速度が所定速度以上であるかどうかを判断する第2判断手段と、前記第1判断手段により前記各導体ラインの接触が保持されている時間が所定時間以下であると判断され、かつ、前記第2判断手段により前記接触位置の移動速度が所定速度以上であると判断された場合、前記ディスプレイ上でカーソルを所定距離だけ移動させるに要する時間を算出する時間算出手段と、前記カーソルを所定距離だけ移動させる毎に前記時間算出手段により算出された時間を一定の割合で増加させる時間増加手段とを備え、前記カーソル移動手段は、前記時間増加手段に増加された時間にて前記カーソルを所定距離だけ移動させ、前記時間増加手段により増加された時間が所定時間以上になった場合に、前記カーソルをディスプレイ上で停止させる停止手段をさらに備えた構成とされる。また、更に、各種データを入力するキーボードを有し、前記パネルスイッチはキーボードに配設された構成とされる。

【0008】

【作用】前記構成を有する本発明では、キーボードに配設されたパネルスイッチ上で指等を介して押下操作が行なわれた場合、パネルスイッチにおける各導体ラインの接触位置が押下開始位置から押下終了位置まで移動されることに対応して、ディスプレイ上に表示されたカーソルが移動される。このとき、第1判断手段を介して、押下終了位置にて各導体ラインの接触が保持されている時間が所定時間以下であるかどうか判断され、また同時に、第2判断手段を介して、各導体ラインの接触位置が押下終了位置の近傍で移動する際における移動速度が所定速度以上であるかどうか判断される。

【0009】そして、第1判断手段により各導体ラインの接触が保持されている時間が所定時間以下であると判断され、かつ、第2判断手段により接触位置の移動速度が所定速度以上であると判断された場合には、時間算出手段を介して、ディスプレイ上でカーソルを所定距離だけ移動させるに要する時間が算出される。この後、カーソルを所定距離だけ移動させ、このようにカーソルを所定距離だけ移動させる毎に、時間増加手段を介して時間算出手段により算出された時間を一定の割合で増加させる。そして、カーソル移動手段を介して、時間増加手段により増加された時間に従ってカーソルが所定距離だけ移動される。また、その増加された時間が所定の時間以上になった場合、カーソルは停止手段を介してディスプレイ上で停止されるものである。

【0010】

【実施例】以下、本発明の一実施例に基づき図面を参照しつつ詳細に説明する。図1は本実施例に係るカーソル移動制御装置の斜視図であり、かかるカーソル移動制御装置1は、カーソルや各種データを表示可能なCRTディスプレイ2、各種キーが配設されたキーボード3、及び、キーボード3の側部(図1中、右側部)に出入可能に配設されたパネルスイッチ4とから基本的に構成される。

【0011】パネルスイッチ4はカーソルを移動させる際に指等にて押下接触され、その押下開始位置から押下終了位置までの移動距離(パネルスイッチ4でオン状態が連続している間に指等がパネルスイッチ4上を移動する距離)に対応して、CRTディスプレイ2上に表示されたカーソルを移動させるものである。次に、かかるパネルスイッチ4の構成について図2、図3を参照して説明する。図2はパネルスイッチ4の一部を構成するフィルム基板の平面図、図3は2つのフィルム基板を重ね合わせてパネルスイッチ4を構成した状態を示す平面図である。図2において、フィルム基板5はポリエステルフィルム基材6からなり、このフィルム基材6の一面上には銀ペースト等の導電性塗料により印刷形成された複数の導電ライン7(図3に示すように本実施例では14個の導電ライン7が形成されている)が設けられている。

5

また、各導電ライン7の間には、そのライン方向に沿って複数個のドットスペーサ8が形成されており、かかる各ドットスペーサ8は、後述するように、フィルム基板5において導電ライン7が形成された面ともう1つのフィルム基板5において導電ライン7が形成された面とを重ね合わせた際に、両フィルム基板5の導電ライン7相互が接触しないように一定の間隔だけ離間した状態で対向配置させるものである。

【0012】前記のように構成された一対のフィルム基板5（一方のフィルム基板5にはドットスペーサ8が設けられていない）は、図3に示すように、各導電ライン7が相互に対向し、且つ、各導電ライン7が相互に直交して縦方向の導電ライン7（下側のフィルム基板5に形成されている）と横方向の導電ライン7（上側のフィルム基板5に形成されている）とがマトリックスを形成するように重ね合わせられる。これにより、パネルスイッチ4が構成されるものである。このように構成されたパネルスイッチ4では、一方のフィルム基板5における各導電ライン7からスイッチ検知信号をシリアルに入力するとともに、他方のフィルム基板5における各導電ライン7から出力されるスイッチ検知信号を検出することにより、パネルスイッチ4上において縦方向の導電ライン7と横方向の導電ライン7との接触ポイント（各導電ライン7は、ドットスペーサ8の作用を介して相互の交差ポイントで接触され、パネルスイッチ4が指等で押下された場合には複数個の交差ポイントでオンされる）を検出するものである。従って、パネルスイッチ4上で指等により押下接触され、押下位置が押下開始位置から押下終了位置まで移動した場合には、各縦方向の導電ライン7と各横方向の導電ライン7との接触ポイントが指等の移動に対応して移動されることとなり、かかる接触ポイントの移動状態が一定の時間間隔で検出されるものである。尚、パネルスイッチ4における接触ポイントの検出方法については公知のものであるので、ここではその説明を省略する。

【0013】次に、カーソル移動制御装置1の制御系について図4に基づき説明する。図4はカーソル移動制御装置1の制御ブロック図を示し、制御装置Cを核として構成される。制御装置Cは、バスラインBを介して相互に接続された中央処理装置（以下、CPUと称する）10、RAM11、ROM12、X方向タイマ13、Y方向タイマ14を備えている。ここに、CPU10はROM12に記憶されている各種の制御プログラムに基づいて各種データの演算を行なうものであり、また、RAM11はCPU10により演算されたデータを一時的に記憶するものである。かかるRAM11には、指等をパネルスイッチ4上で押下しつつ移動した際に、パネルスイッチ4上で検出された複数個（本実施例では9個）の各接触ポイント（各接触ポイントは前記のようにパネルスイッチ4を指等で押下した際に各導電ライン7の交差ポ

6

イントでオンされる）のそれぞれについてX座標データ、Y座標データ、移動時間データ、後述する包含に関するデータ等を記憶するリングバッファが設けられている。また、RAM11には、各接触ポイントについてROM12に記憶されたスキニングプログラムに基づいてパネルスイッチ4における各導電ライン7の交差ポイントをスキャンした際に、前回スキャン時においてオンされた複数個の各接触ポイントの位置と数（前回スキャン時のオン領域）、及び、今回スキャン時においてオンされた複数個の各接触ポイントの位置と数（今回スキャン時のオン領域）を記憶するオン領域メモリ等のメモリが設けられている。

【0014】また、ROM12には、図4に示すように、後述する包含値、及び、各接触ポイントの始点グループと終点グループとの間における距離とからリングバッファ内における始点-終点間距離を求めるためのテーブルが記憶されており、更に、図5に示すように、後述する終点グループにおける各接触ポイントの平均移動速度、及び、実際の押下開始位置における接触ポイントと押下終了位置における接触ポイントとの間の距離に基づいて、X方向、Y方向についてのカーソルの移動速度を求めるためのテーブルが記憶されている。また、ROM12には、後述するカーソル移動制御の基本プログラム、パネルスイッチ4上で指等が弾き上げ動作を伴いつつ押下動作が行なわれたかどうかを判断する弾き上げ動作検出プログラム、弾き上げ動作時にカーソルの移動を制御するカーソル移動プログラム、X方向タイマ13の割り込み処理プログラム、Y方向タイマ14の割り込み処理プログラム、その他カーソル移動制御装置1の制御上必要な各種のプログラムが記憶されている。

【0015】更に、X方向タイマ13は、弾き上げ動作時にカーソルのX方向における移動を行なう際に起動されて割り込み処理を行なうためのタイマであり、また、Y方向タイマ14は、同様に弾き上げ動作時にカーソルのY方向における移動を行なう際に起動されて割り込み処理を行なうためのタイマである。

【0016】また、制御装置Cには出力ポート15及び入力ポート16を介してパネルスイッチ4が接続されており、これよりCPU10は、ROM12に記憶されたスキニングプログラムに基づいて、出力ポート15からスイッチ検知信号をパネルスイッチ4における一方のフィルム基板5の各導電ライン7に出力し、他方のフィルム基板5の各導電ライン7から出力されるスイッチ押下信号（押下されている接触ポイントから出力される）を入力ポート16から取り込む。これにより、CPU10は、パネルスイッチ4上において押下接触されている各接触ポイントの位置（X座標データ、Y座標データ）を検出してメモリに記憶する。更に、制御装置Cには、出力ポート17を介してCRTディスプレイ2が接続されており、CPU10は後述のカーソル移動制御プログ

ラムに基づいてディスプレイ2上に表示されたカーソルの移動制御を行なうものである。

【0017】続いて、前記のように構成されたカーソル移動制御装置1の動作について図7乃至図13に基づき説明する。ここで、パネルスイッチ4は、図12に押下操作状態を示すように、指20により押下開始位置Aから押下終了位置Bまで軌跡Zをもって押下操作されたものとする。また、かかる押下操作の間におけるパネルスイッチ4上でオンされた各ポイントP1乃至P12と各種情報との関係は、図13に示すような関係を有しているものとする。即ち、図13はパネルスイッチ4上でオンされた各ポイントP1乃至P12と各種情報との関係を模式的に示す説明図であり、図13中、各オンポイントP1乃至P12は、指20にてパネルスイッチ4を押下した時には複数個のポイントが同時にオンされることから、オンされた各ポイントのX座標データとY座標データの平均値を求めて各ポイントを代表するポイントとして表わされており、本実施例では4ポイントの平均値を求めて表わしている。これらの各ポイントは、CPU10によるスキニング動作を介して5msec毎に検出される。

【0018】また、パネル移動箇所数Cは、2個の縦方向導電ライン7間又は2個の横方向導電ライン7間の距離を1ステップとした場合に、前記押下操作中に横切ったステップ数を表わしている。また、バッファ内データDは、前記したように、RAM11のリングバッファ内に記憶されている9個のポイント(押下終了位置BにおけるポイントP12から遡って9個のポイント)に関するデータを意味する。更に、各ポイントP1乃至P12の内、ポイントP12から押下開始方向に遡って4個のポイント(P12、P11、P10、P9)を終点グループEGとし、各4個のポイントのX座標データとY座標データとの平均値により表わされるポイントF(星形マークにて示す)を終点グループEGの代表値とする。また、ポイントP8から押下開始方向に遡って4個のポイント(P8、P7、P6、P5)を始点グループSGとし、各4個のポイントのX座標データとY座標データとの平均値により表わされるポイントH(星形マークにて示す)を始点グループSGの代表値とする。更に、このように求めた終点グループ代表値Fと始点グループSGとの間の距離をJ(ステップ数で表わされる)とする。

【0019】先ず、カーソル移動制御の基本動作について図7に基づき説明する。図7はカーソル移動制御における基本プログラムのフローチャートを示し、先ず、ステップ(以下、Sと略記する)1において、パネルスイッチ4が押下されているかどうか判断される。押下されていない場合(S1:NO)には押下されるまで待機する一方、押下されている場合(S1:YES)にはパネルスイッチ4上の押下位置までカーソルを移動させる

(S2)。そして、再度パネルスイッチ4が押下されているかどうか判断され(S3)、パネルスイッチ4が押下されている限り(S3:YES)S2に戻ってカーソルを移動させる。一方、S3にてパネルスイッチ4の押下が終了したと判断された場合(S3:NO)には、後述する弾き上げ動作をもってパネルスイッチ4が押下されたかどうか判断される(S4)。弾き上げ動作の条件を満たさない場合(S4:NO)、カーソル移動を停止した(S5)後S1に戻る。また、弾き上げ動作条件を満たす場合(S4:YES)には、その弾き上げ動作に従って後述するカーソルの連続移動を行う(S6)。この後、パネルスイッチ4が押下されているかどうか判断され(S7)、押下されていない場合(S7:NO)にはS6に戻る一方、押下されている場合(S7:YES)にはカーソルの連続移動を停止した(S8)後、S1に戻る。

【0020】このように、S6において、カーソルの連続移動が行なわれている間にパネルスイッチ4の押下された場合には、カーソルの連続移動の停止が行なわれることから、カーソルを迅速に移動させつつ所望の位置で停止させることが可能となるものである。

【0021】次に、前記S4において行なわれる弾き上げ動作条件の判断処理、前記S6にて行なわれるカーソルの連続移動処理について図8乃至図11に基づき説明する。先ず、前記S4にて行なわれる弾き上げ動作条件の判断処理について図8を参照して説明する。図8は弾き上げ動作条件の検出プログラムのフローチャートであり、S10にて前記した始点グループSGと終点グループEGとの間のグループ間距離Jが計算される。そして、その計算されたグループ間距離Jが2ステップよりも小さいかどうか判断される(S2)。このような判断を行なうのは、グループ間距離Jが2ステップよりも小さい程度の場合には通常のスปีドによる入力動作と考えられ、また、グループ間距離Jが2ステップよりも大きい場合には弾き上げ動作が行なわれた確率が高いと考えられることに基づく。従って、グループ間距離Jが2よりも小さい場合(S11:YES)には、弾き上げ動作は行なわれなかったものと判断して処理をリターンする。一方、グループ間距離Jが2ステップ以上である場合(S11:NO)にはS12に移行する。

【0022】S12においては、前記押下開始位置AのポイントP1と押下終了位置BのポイントP12との間の距離が計算され、かかる距離が3ステップよりも小さいかどうか判断される(S13)。かかる判断は、前記S12におけると同様、押下開始位置AのポイントP1と押下終了位置BのポイントP12との間の距離が3ステップ以下の場合には通常のスปีドをもって押下操作されたものと考えられるのに対して、かかる距離が3ステップ以上の場合には弾き上げ動作をもって押下操作が行なわれた確率が高いと考えられるからである。従っ

て、ポイントP1とポイントP12間の距離が、3ステップ以下の場合(S13: YES)には弾き上げ動作が行なわれなかったものとして処理をリターンする。一方、各ポイントP1、P12間距離が3ステップ以上の場合(S13: YES)にはS14に移行する。

【0023】S14においては、最終ポイントP12から押下開始方向に遡って5ポイント(ポイントP12、P11、P10、P9、P8)の内、1ステップ移動するのに要した時間が一定値を越した回数が計算される。ここに、一定値としては32 msecが設定されている。この後、このように計算された回数が0回以下であるかどうか判断される(S15)。かかるS15は、押下終了位置Bの近傍において前記各ポイントの移動速度を観測することにより、弾き上げ動作が行なわれたかどうかを判断するためのものである。即ち、弾き上げ動作が行なわれる場合には、押下終了位置Bの近傍において各ポイントの移動速度は速くなり、各ポイントが1ステップを移動するための時間は一定時間(32 msec)以下と考えられ、これに対して通常のスピードによる押下操作の場合には、押下終了位置Bの近傍で各ポイントの移動速度が低下して1ステップを移動するのに一定時間よりも長い時間を要すると考えられるからである。従って、前記S14にて計算された回数が0以上である場合(S15: YES)には、弾き上げ動作が行なわれなかったものとして処理がリターンされ、また、0以下である場合(S15: NO)には、弾き上げ動作が行なわれたものとしてS16に移行する。

【0024】S16では、押下終了位置BのポイントP12がオンされた時間が計算され、その計算された時間が設定値(32 msec)よりも大きいかどうか判断される。この判断は、弾き上げ動作が行なわれた場合には指20が速く動作することから最終ポイントP12がオンされている時間が短く、また、通常の押下動作の場合には指20が最終ポイントP12で比較的長い時間留まっていることを考慮したものである。従って、ポイントP12のオン時間が設定値よりも長い場合(S16: YES)には弾き上げ動作が行なわれなかったものとして処理がリターンされ、一方、ポイントP12のオン時間が設定値よりも短い場合(S16: YES)には弾き上げ動作が行なわれたものとしてS18に移行する。

【0025】S18においては、押下終了位置BのポイントP12から押下開始位置方向に遡って5ポイント(P12、P11、P10、P9、P8)のそれぞれに関して、RAM11のメモリに記憶されている前回スキャン時のオン領域(同時にオンされたポイント数、以下同じ)と今回スキャン時のオン領域とのラップ状態を判断される。即ち、ここでは前回スキャン時のオン領域が今回スキャン時のオン領域を包含する回数の値を包含値として、その包含値の計算が行なわれる。ここに、前記のように定義される包含値という概念を導入したの

は、弾き上げ動作が行なわれる場合には各ポイントの移動が速く行なわれることから、包含値は一定値以下になることが考えられ、一方、通常の押下動作が行なわれる場合には各ポイントの移動がゆっくり行なわれることから、包含値は一定値以上になると考えられるからである。

【0026】そこで、S19においては、前記のように計算された包含値が5に等しいかどうか判断される。ここで、包含値が5に等しいということは、パネルスイッチ4を押下した指がほぼ垂直方向に離されたことを意味する。従って、包含値が5である場合(S19: YES)には弾き動作が行なわれなかったものとして処理がリターンされる。これに対して、包含値が5でない場合(S19: NO)には弾き上げ動作が行なわれたものとしてS20に移行する。続く、S20では、ROM12に記憶されている図5のテーブルに基づいて、前記のように計算された包含値、及び、終点グループEGの終点グループ代表値Fと始点グループSGの始点グループ代表値Hとの間の距離Jから、RAM11のリングバッファ内に記憶されたバッファ内データDにおける始点(ポイントP4)と終点(ポイントP12)との間の距離が求められる。例えば、包含値が2で、グループ間距離Jが3の場合には、バッファ内データの始点-終点間距離は4となる。

【0027】この後、S21では、リングバッファに記憶された始点(ポイントP4)のX、Y座標データと終点(ポイント12)のX、Y座標データとに基づいて、リングバッファ内における実際の始点-終点間距離が計算され、その計算された距離と前記S20にてテーブルから求めたテーブル値とが比較される。そして、バッファ内始点-終点間距離がテーブル値よりも小さい場合(S21: YES)には、弾き上げ動作が行なわれていないものとして処理がリターンされる。一方、バッファ内始点-終点間距離がテーブル値よりも大きい場合(S21: NO)には、弾き上げ動作に基づいて、後述するカーソルの移動処理が行なわれる(S22)。

【0028】次に、パネルスイッチ4が弾き上げ動作を伴って押下操作された場合、前記S22にて行なわれるカーソルの移動処理について図9を参照して説明する。図9は弾き上げ動作に基づくカーソルの移動制御プログラムのフローチャートであり、先ずS30にて終点グループEGのX方向平均移動時間が計算される。かかるX方向平均移動時間は、RAM11のリングバッファに記憶されている各ポイント間の移動時間に基づいて計算されるものである。また、これと同時に、終点グループEGのX方向(図13参照)における移動量が求められる。かかる移動量は終点グループEGの終点グループ代表値Fと始点グループSGの代表値HとのX方向における距離から求められ、因みに、前記距離が0の場合には、X方向移動量は0となり、前記距離が0以外の場合

には、X方向移動量は1となる。

【0029】続いて、S31において、パネルスイッチ4上における押下開始位置AのポイントP1のX座標データと押下終了位置BのポイントP12のX座標データとから、各ポイントP1、P12間のX方向の距離が計算され、前記S30で計算されたX方向平均移動時間と各ポイントP1、P12間距離とに基づいて、ROM12に記憶されている図6のテーブルからCRTディスプレイ2上でカーソルを移動させるためのX方向移動時間が求められる。かかるX方向移動時間はX方向タイマ13にセットされる。

【0030】前記S30、S31と同様にして、終点グループEGの各ポイントについて、Y方向の平均移動速度、及び、Y方向の移動量が計算され(S32)、更に、各ポイントP1、P12間のY方向の距離が計算されるとともに、Y方向平均移動速度と各ポイントP1、P12間の距離とに基づいて、図6のテーブルからカーソルを移動させるためのY方向移動時間が求められる(S33)。かかるY方向移動時間はY方向タイマ14にセットされる。

【0031】この後、S34にて前記S30にて求めたX方向移動量が0であるかどうか判断され、X方向移動量が0である場合(S34: YES)にはS36に移行する。一方、X方向移動量が0でない場合(S34: NO)には、X方向タイマ13を起動して後述の割り込み処理が可能な状態とされる(S35)。続くS36においては、S32にて求めたY方向移動量が0であるかどうか判断され、Y方向移動量が0である場合(S36: YES)には処理がリターンされる一方、Y方向移動量が0でない場合(S36: NO)にはY方向タイマ14を起動して後述の割り込み処理が可能な状態とされた(S37)後、処理がリターンされる。

【0032】次に、X方向タイマ13を介して行なわれる割り込み処理について図10に基づき説明する。図10はX方向タイマ13によるタイマ割り込み処理プログラムのフローチャートであり、先ずS40にて、X方向タイマ13にセットされたX方向移動時間に基づいて、カーソルがCRTディスプレイ2上においてX方向に1ステップ移動される。続くS41では、次回にカーソルを移動させる際にX方向タイマ13にセットされる時間が、前回のタイマセット時間(S31で求めた時間)とそのタイマセット時間に2-7を乗じた時間とを加えた時間が計算される。そして、このように計算されたタイマセット時間と所定の設定値(本実施例では38 msecに設定されている)との比較が行なわれる(S42)。比較の結果、S41にてX方向タイマ13にセットされた時間が設定値よりも小さい場合(S42: NO)、更にCRTディスプレイ2上でカーソルを移動させるべくX方向タイマ13に前記S41で計算された時間が再設定された後、X方向タイマ13が再起動されて

処理がリターンされる。一方、前記S42において、S41にて計算されたタイマセット時間が設定値よりも大きい場合(S42: YES)には、X方向タイマ13が停止されるとともに、カーソルの移動も停止される。この後、処理はリターンされる。

【0033】続いて、Y方向タイマ14を介して行なわれる割り込み処理について図11に基づき説明する。図11はY方向タイマ14によるタイマ割り込み処理プログラムのフローチャートであり、基本的には前記図11に示すX方向タイマの割り込み処理と同様の処理が行なわれる。先ずS50にて、Y方向タイマ14にセットされたY方向移動時間に基づいて、カーソルがCRTディスプレイ2上においてY方向に1ステップ移動される。続くS51では、次回にカーソルを移動させる際にY方向タイマ14にセットされる時間が、前回のタイマセット時間(S33で求めた時間)とそのタイマセット時間に2-7を乗じた時間とを加えた時間が計算される。そして、このように計算されたタイマセット時間と所定の設定値(本実施例では38 msecに設定されている)との比較が行なわれる(S52)。比較の結果、S51にてY方向タイマ14にセットされた時間が設定値よりも小さい場合(S52: NO)、更にCRTディスプレイ2上でカーソルを移動させるべくY方向タイマ14に前記S51で計算された時間が再設定された後、Y方向タイマ14が再起動されて処理がリターンされる。一方、前記S52において、S51にて計算されたタイマセット時間が設定値よりも大きい場合(S52: YES)には、Y方向タイマ14が停止されるとともに、カーソルの移動も停止される。この後、処理はリターンされる。

【0034】前記各X方向タイマ13、Y方向タイマ14による割り込み処理を行なうことにより、パネルスイッチ4上で弾き上げ動作を伴って押下操作が行なわれた場合には、パネルスイッチ4上における指20の移動量が小さい時でもカーソルがCRTディスプレイ2上で慣性的に移動されることとなり、操作者の操作感覚に合致させながらカーソルをCRTディスプレイ2上で迅速、且つ、飛躍的に移動させ得るものである。

【0035】以上詳細に説明した通り本実施例に係るカーソル移動制御装置1では、パネルスイッチ4上において指等により押下操作された場合、その押下操作が通常の押下動作により行なわれたものであるか、又は、弾き上げ動作により行なわれたものであるかを判断し、弾き上げ動作を伴って押下操作が行なわれたと判断された場合には終点グループEGにおける各ポイントP12乃至P8のX方向及びY方向の平均移動速度に基づいて各X方向タイマ13、Y方向タイマ14にセットされるタイマセット時間を漸増させつつ割り込み処理を行なうとともに、タイマセット時間が所定の設定値以下になった場合にCRTディスプレイ2上でカーソルを停止させるよ

うにしたので、パネルスイッチ4の押下移動量が小さい場合においても、パネルスイッチ4上で弾き上げ動作を伴う押下操作に基づいてカーソルをCRTディスプレイ2上で迅速、且つ、飛躍的に移動させることができる。これにより、操作者のパネル操作感覚に合致させながら、カーソル移動の操作性を格段に向上することができるものである。

【0036】また、本実施例に係るパネルスイッチ4は、キーボード3に対して出入可能に配設されているので、パネルスイッチ4のために特にスペースを設ける必要がなく、従って、従来のカーソル制御装置に比して小型軽量化を図ってコンパクト化を達成することができるものである。尚、本発明は前記実施例に限定されるものではなく、本発明の要旨を逸脱しない範囲内で種々の改良、変形が可能であることは勿論である。

【0037】

【発明の効果】以上説明した通り本発明は、小型化を図ることによりキーボード等の入力装置に配設可能とするとともに、操作者がパネルスイッチ上で指等を移動する際に、パネルスイッチ上で指等をスライドさせる通常の

【図面の簡単な説明】

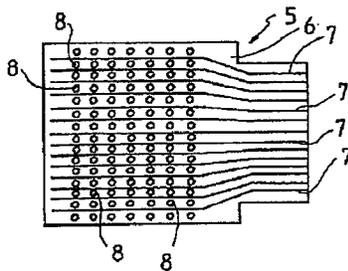
【図1】カーソル移動制御装置の斜視図である。

【図2】パネルスイッチの一部を構成するフィルム基板の平面図である。

【図3】2つのフィルム基板を重ね合わせてパネルスイッチを構成した状態を示す平面図である。

【図4】カーソル移動制御装置の制御ブロック図である。

【図2】



*【図5】リングバッファ内における始点-終点間距離を求めるためROMに記憶されたテーブルである。

【図6】カーソルの移動速度を求めるためROMに記憶されたテーブルである。

【図7】カーソル移動制御における基本プログラムのフローチャートである。

【図8】弾き上げ動作条件の検出プログラムのフローチャートである。

10 【図9】弾き上げ動作に基づくカーソルの移動制御プログラムのフローチャートである。

【図10】X方向タイマによるタイマ割り込み処理プログラムのフローチャートである。

【図11】Y方向タイマによるタイマ割り込み処理プログラムのフローチャートである。

【図12】パネルスイッチの押下操作状態を示す説明図である。

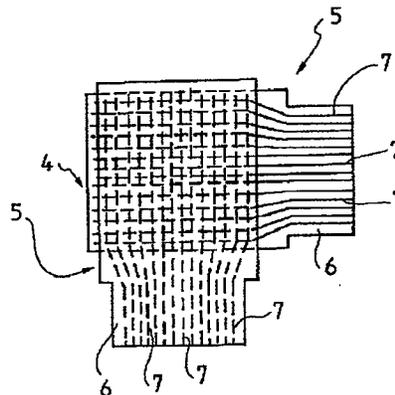
【図13】パネルスイッチ上でオンされた各ポイントと各種情報との関係を模式的に示す説明図である。

【符号の説明】

- 1 カーソル移動制御装置
- 2 CRTディスプレイ
- 3 キーボード
- 4 パネルスイッチ
- 5 フィルム基板
- 7 導電ライン
- 8 ドットスベーサ
- 10 CPU
- 11 RAM
- 12 ROM
- 30 13 X方向タイマ
- 14 Y方向タイマ
- A 押下開始位置
- B 押下終了位置

*

【図3】

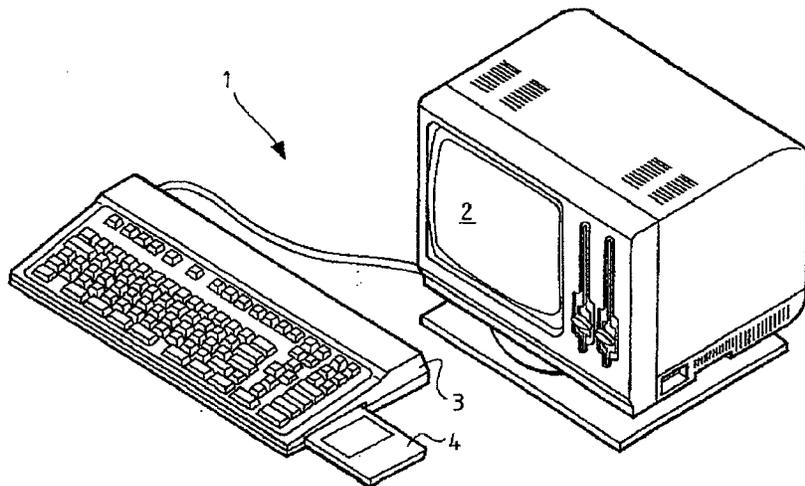


【図5】

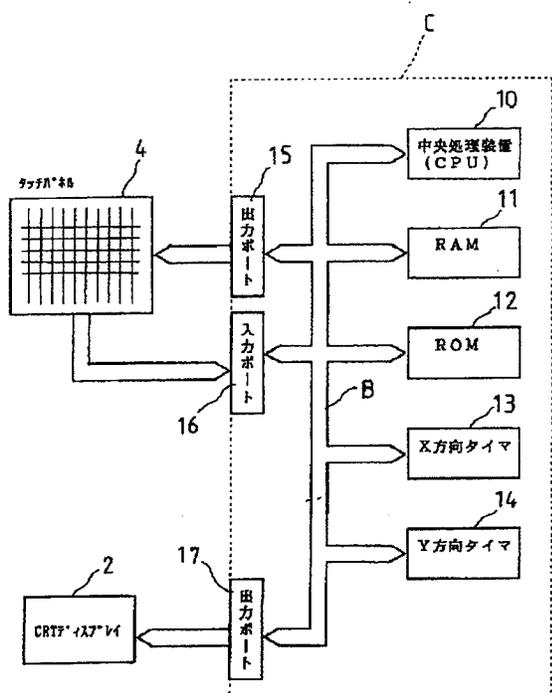
バッファ内データ始点-終点間距離

		包含値					
		0	1	2	3	4	5
始点-終点間距離	7	8	8	8	8	8	X
	6	7	7	7	7	7	X
	5	6	6	6	6	6	X
	4	5	5	5	5	6	X
	3	4	4	4	4	5	X
	2	3	3	3	3	5	X
1	X	X	X	X	X	X	
0	X	X	X	X	X	X	

【図1】



【図4】

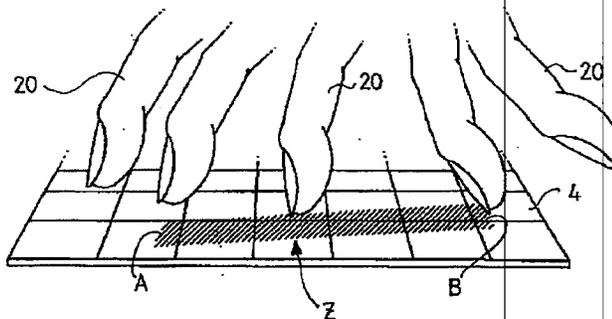


【図6】

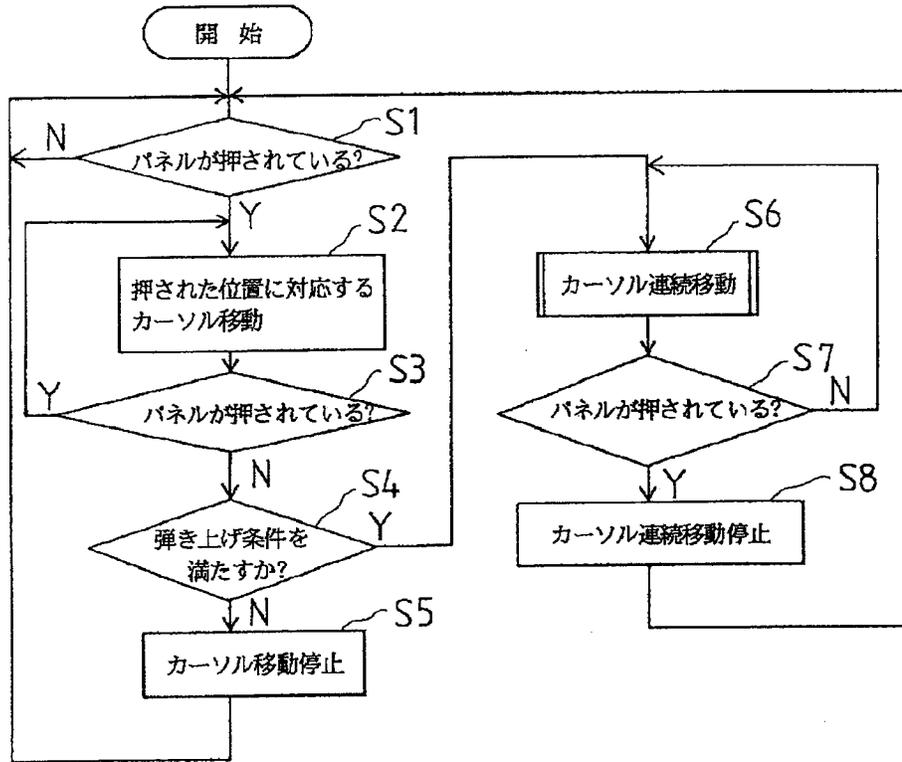
カーソル移動時間

開始点	終点-終点間距離														
	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29
6	2	2	3	4	5	6	6	6	6	6	6	6	6	6	6
12	4	4	5	6	7	8	8	8	8	8	8	8	8	8	8
18	8	9	10	11	12	13	14	14	14	14	14	14	14	14	14
22	18	19	21	23	26	30	34	38	42	46	50	54	58	62	66
26	24	26	28	32	36	42	48	54	60	66	72	78	84	90	96
30	46	50	56	62	68	74	80	86	92	98	104	110	116	122	128
36	68	76	84	90	96	102	108	114	120	126	132	138	144	150	156
68	80	98	110	124	140	160	180	200	220	240	260	280	300	320	340
80	150	180	200	220	240	260	280	300	320	340	360	380	400	420	440

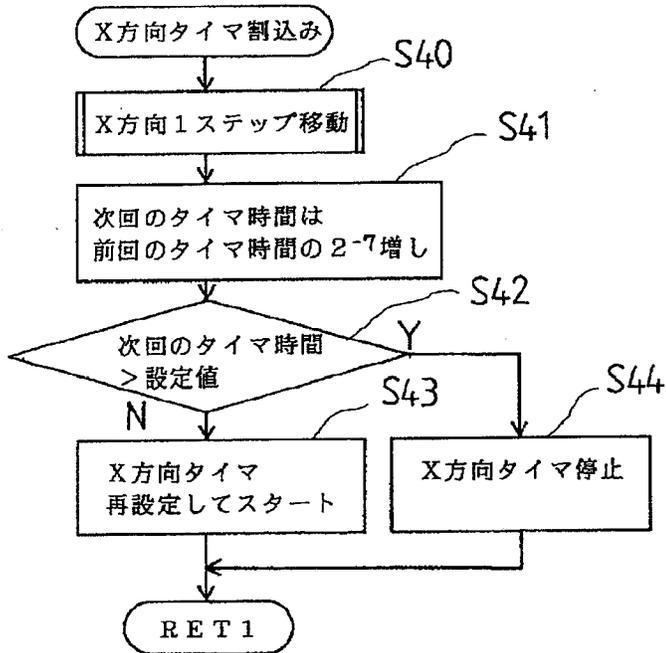
【図12】



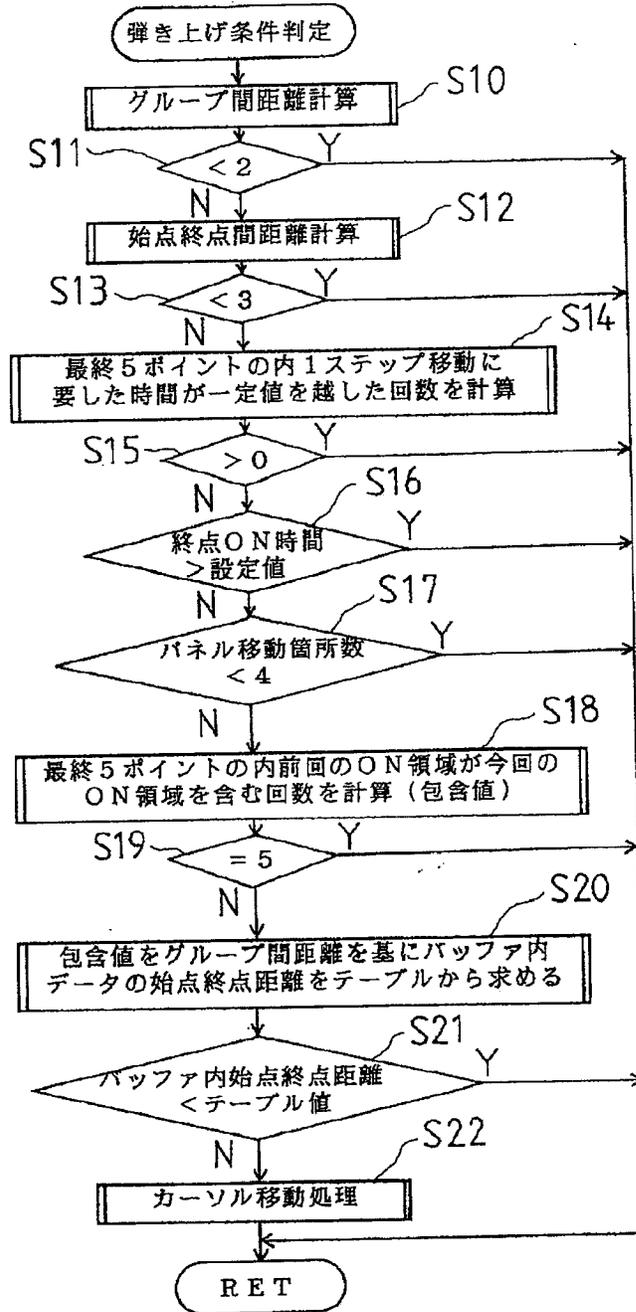
【図7】



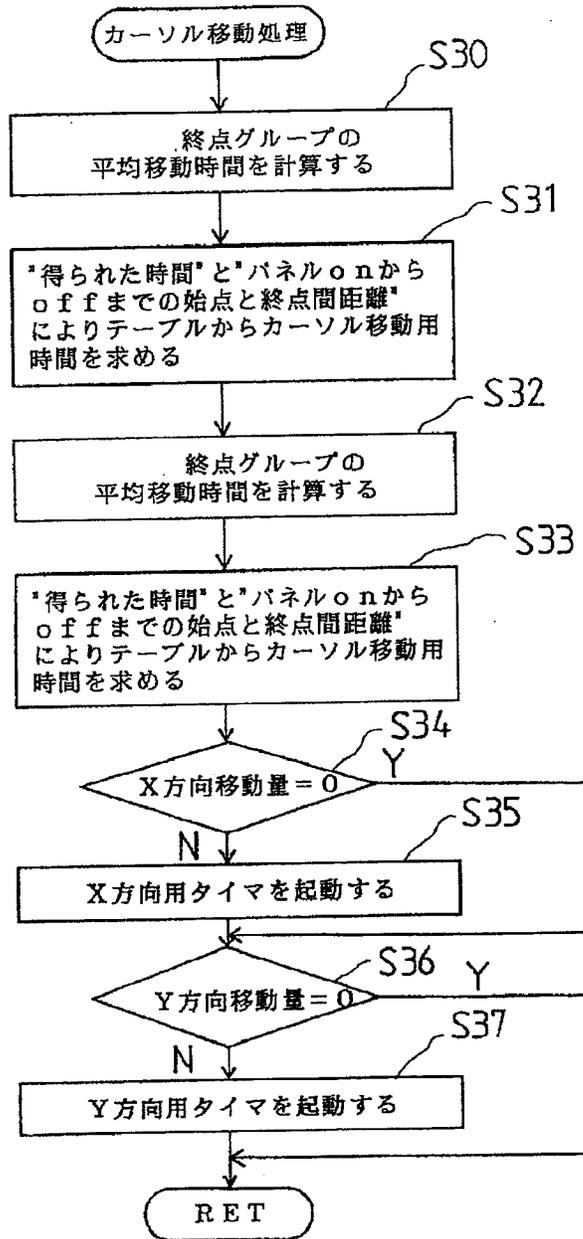
【図10】



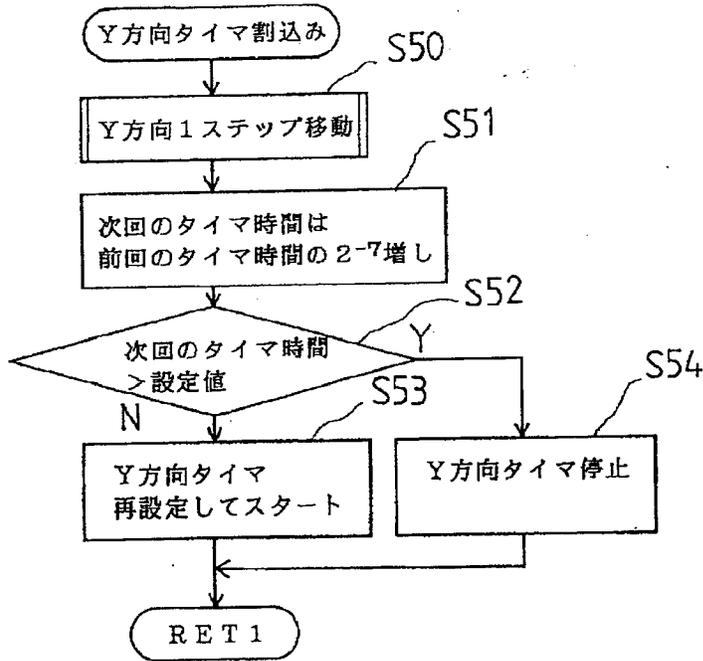
【図8】



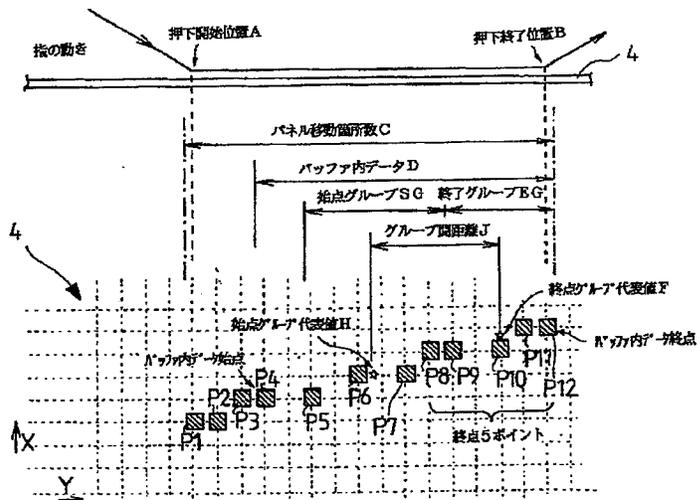
【図9】



【図11】



【図13】



WORK-SURFACE SYSTEM AND REALIZING METHOD THEREOF, SENSOR UTILIZING METHOD AND PERSON IDENTIFYING METHOD

Patent number: JP6083523 (A)

Publication date: 1994-03-25

Inventor(s): TOOMASU EICHI SUPITAA +

Applicant(s): AMERICAN TELEPHONE & TELEGRAPH +

Classification:

- international: G06F3/047; G06F3/033; G06F3/041; G06F3/048; G06K9/00; G06T7/00; G07C9/00; G06F3/033; G06F3/041; G06F3/048; G06K9/00; G06T 7/00; G07C9/00; (IPC1-7): G06F3/03; G06F3/03; G06F15/62

- european: G06F3/041; G06F3/033; G06F3/041F; G06K9/00H; G07C9/00C2D

Application number: JP199203524 27 19921211

Priority number(s): US19910807524 19911213

Also published as:

EP0546704 (A2)

EP0546704 (A3)

JP9069021 (A)

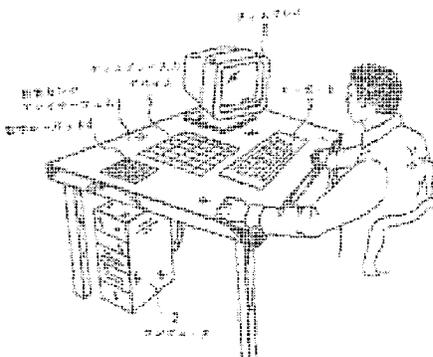
CA2203297 (A1)

CA2078607 (A1)

Abstract of JP 6083523 (A)

PURPOSE: To provide a method and device for realizing an intelligent work surface.

CONSTITUTION: A work surface such as a disk top or a floor is covered with a tactile sensing array 1 which supplies tactile sense data to a processor 2. An area on the sensing array 1 is normalized so that the prescribed segment element of the sensing array 1 can be included, and set so as to be functioned as an input device 5 such as a keyboard 3 or a mouse. The area can be re-normalized even after normalized in order to rearrange the specific input device. The area can be re-normalized so as to be functioned as a different input device. The array 1 collects data in a prescribed interval for the purpose of analysis such as the identification of an individual or an object. For example, the image of force such as the hand print or foot print of the individual is collected, analyzed, and compared with data indicating the image of the force corresponding the known individual.



Data supplied from the *espacenet* database — Worldwide



(12) **EUROPEAN PATENT APPLICATION**

(21) Application number: **92310595.1**

(51) Int. Cl.⁵: **G06F 3/033, G07C 9/00, A61B 5/117**

(22) Date of filing: **20.11.92**

A request for correction of the numbering of claims number 29 as to read number 28 has been filed pursuant to Rule 88 EPC. A decision on the request will be taken during the proceedings before the Examining Division (Guidelines for Examination in the EPO, A-V, 2.2).

(71) Applicant: **AMERICAN TELEPHONE AND TELEGRAPH COMPANY**
32 Avenue of the Americas
New York, NY 10013-2412(US)

(72) Inventor: **Speeter, Thomas H.**
429 Ely Harmony Road
Freehold, New Jersey 07728(US)

(30) Priority: **13.12.91 US 807524**

(43) Date of publication of application:
16.06.93 Bulletin 93/24

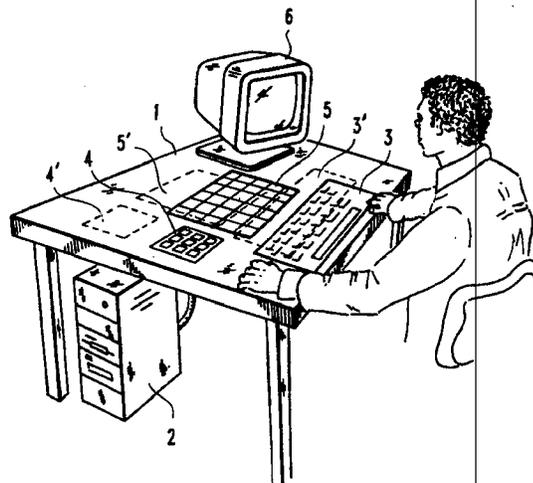
(74) Representative: **Johnston, Kenneth Graham et al**
AT&T (UK) Ltd. 5 Mornington Road
Woodford Green Essex, IG8 OTU (GB)

(84) Designated Contracting States:
DE FR GB IT SE

(54) **Intelligent work surfaces.**

(57) Methods and apparatus for implementing intelligent work surfaces are provided. Work surfaces, such as desktops and floors, are covered with a tactile sensing array to provide tactile data to a processor. Regions on the sensing arrays may be defined to include predetermined sensing elements of the sensing array and may be specified to function as input devices, such as a keyboard, a mouse, or the like. Once defined, regions may be redefined to reposition a specified input device. Regions also may be respecified to function as a different input device. The arrays also may collect data at predetermined intervals for analysis, such as for identifying an individual or an object. For example, the force-image of an individual's handprint or footprint could be taken, analyzed, and compared against data representing the force-images of known individuals.

FIG. 2



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平6-83523

(43)公開日 平成6年(1994)3月25日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 3/03	3 8 0 H	7165-5B		
	C	7165-5B		
	3 1 5 B	7165-5B		
15/62	4 6 5 K	9071-5L		

審査請求 有 請求項の数24(全 17 頁)

(21)出願番号 特願平4-352427

(22)出願日 平成4年(1992)12月11日

(31)優先権主張番号 8 0 7 5 2 4

(32)優先日 1991年12月13日

(33)優先権主張国 米国 (U S)

(71)出願人 390035493
 アメリカン テレフォン アンド テレグラ
 フ カムパニー
 AMERICAN TELEPHONE
 AND TELEGRAPH COMPA
 NY
 アメリカ合衆国 10013-2412 ニューヨ
 ーク ニューヨーク アヴェニュー オブ
 ジ アメリカズ 32

(72)発明者 トーマス エイチ. スピーター
 アメリカ合衆国 07728 ニュージャージ
 ー フリーホルド、エリー ハーモニー
 ロード 429

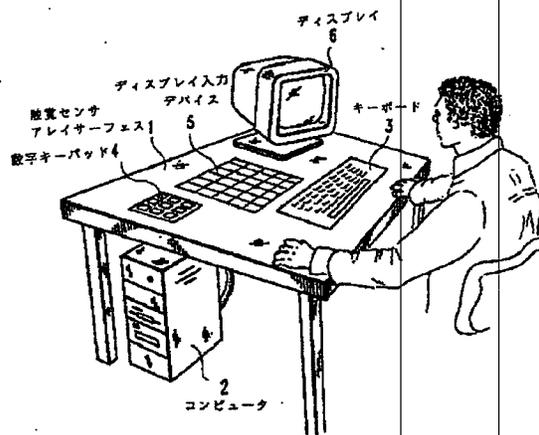
(74)代理人 弁理士 三俣 弘文

(54)【発明の名称】 ワークサーフェスシステム及びその実現方法、及びセンサ利用方法及び人物識別方法

(57)【要約】

【目的】 インテリジェントワークサーフェスを実現する方法及びその装置を提供する。

【構成】 デスクトップ及びフロア等のワークサーフェスが、触覚データをプロセッサに供給する触覚センシングアレイによって覆われる。センシングアレイ上の領域はセンシングアレイの所定のセンシングエレメントを含むように規定され、キーボード、マウスなどの入力デバイスとして機能するように設定される。規定された後も、領域は特定の入力デバイスを再配置するために再規定されうる。領域は相異なった入力デバイスとして機能するように再規定されうる。アレイは、個人あるいは物体を識別するなどの解析目的で所定の間隔でデータを収集する。例えば、個人の手形あるいは足形などの力のイメージが収集され、解析され、既知の個人に対する力のイメージを表すデータと比較される。



1

【特許請求の範囲】

【請求項1】 ワークサーフェスを構成するセンシング
エレメントよりなるアレイと、
当該アレイの各々のセンシングエレメントは、当該セン
シングエレメントへの入力に応答してデータ出力を生成
し、

第一の複数のセンシングエレメントよりなる第一の入
力デバイスを規定する手段と、

前記第一の複数のセンシングアレイからの各々のデー
タ出力を前記第一の入力デバイスの対応する出力にマッ
ピングする手段と、

前記第一の入力デバイスを第二の複数のセンシングエ
レメントよりなるものとして再規定する手段と、 かなり、

前記マッピング手段が、前記第二の複数のセンシング
エレメントからのデータ出力を前記第一の入力デバイス
の対応する出力としてマッピングすることを特徴とする
ワークサーフェスシステム。

【請求項2】 前記第一の複数のセンシングエレメン
トの少なくとも一部からのデータ出力を第二の入力デバ
イスの対応する出力にマッピングする手段を更に有する
ことを特徴とする請求項第1項に記載のシステム。

【請求項3】 第二の入力デバイスを第三の複数のセン
シングエレメントよりなるものとして規定する手段
と、

前記第三の複数のセンシングエレメントからの各々の
データ出力を第二の入力デバイスからの対応する出力に
マッピングする手段とを更に有することを特徴とする請
求項第1項に記載のシステム。

【請求項4】 前記第一の入力デバイスからの出力を連
結領域に分割するために、当該出力に基づいて連結領域
解析を実行する手段を更に有することを特徴とする請求
項第1項に記載のシステム。

【請求項5】 少なくとも一つの連結領域の重心を決定
する手段を更に有することを特徴とする請求項第4項に
記載のシステム。

【請求項6】 前記第一の入力デバイスが、画像インタ
ーフェースを有しており、

前記第一の複数のセンシングエレメントへの入力のマ
ルチレベル表示を画像としてストアする手段を更に有す
ることを特徴とする請求項第1項に記載のシステム。

【請求項7】 前記規定手段が、前記アレイ上で複数の
入力デバイスを同時に規定する手段を有し、

前記複数の入力手段の各々が、前記センシングエレメ
ントの内の所定のものよりなることを特徴とする請求項
第1項に記載のシステム。

【請求項8】 ワークサーフェスを構成するセンシング
エレメントよりなるアレイと、

当該アレイの各々のセンシングエレメントは当該セン
シングエレメントへの入力に応答してデータ出力を生成

2

し、

第一の複数のセンシングエレメントよりなる第一の入
力デバイスを規定する手段と、

前記第一の複数のセンシングアレイからの各々のデー
タ出力を前記第一の入力デバイスの対応する出力にマッ
ピングする手段と、

前記第一の複数のセンシングエレメントの内の少なく
とも一部からのデータ出力を第二の入力デバイスの対応
する出力にマッピングする手段とを有することを特徴と
するワークサーフェスシステム。

10

【請求項9】 各々入力に応答してデータ出力を生成す
る複数のセンシングエレメントを有するセンサを用い
る方法において、

第一の複数の前記センシングエレメントよりなるもの
として第一の入力デバイスを規定するステップと、

前記第一の複数のセンシングエレメントからのデータ
出力を前記第一の入力デバイスの対応する出力にマッピ
ングするステップと、

20

前記第一の入力デバイスを第二の複数の前記センシ
ングエレメントよりなるものとして再規定するステップ
と、

かなり、

前記マッピングステップは、前記第二の複数のセンシ
ングエレメントからの出力を前記第一の入力デバイスの
対応する出力にマッピングすることを特徴とするワーク
サーフェスシステム実現方法。

【請求項10】 前記第一の複数のセンシングエレメ
ントの少なくとも一部からのデータ出力を第二の入力デ
バイスの対応する出力にマッピングするステップを更に
有することを特徴とする請求項第9項に記載の方法。

30

【請求項11】 第二の入力デバイスを第三の複数の
センシングエレメントよりなるものとして規定するステ
ップと、

前記第三の複数のセンシングエレメントからの各々の
データ出力を第二の入力デバイスからの対応する出力に
マッピングするステップとを更に有することを特徴とす
る請求項第9項に記載の方法。

【請求項12】 前記規定ステップが、
センサのセンシングエレメントをサンプリングするステ
ップと、

40

所定のパターンを有する力が前記センサに対して印加さ
れたか否かを決定するステップと、

前記所定のパターンを有する力に応答して所定の入力デ
バイスの位置及び方向を規定するステップとを有する
ことを特徴とする請求項第9項に記載の方法。

【請求項13】 前記第一の入力デバイスからの出力を
連結領域に分割するために当該出力に基づいて連結領域
解析を実行するステップを更に有することを特徴とする
請求項第9項に記載の方法。

50

【請求項14】 少なくとも一つの連結領域の重心を決

定するステップを更に有することを特徴とする請求項第9項に記載の方法。

【請求項15】 原点を前記規定された領域内で第一の力が印加された重心であると規定するステップと、第二の力の重心が前記原点から少なくとも第一の所定の距離だけ離れているか否かを決定するステップと、前記第二の力の重心が前記原点から少なくとも前記第一の所定の距離だけ離れている場合に、ディスプレイスクリーン上のカーソルの所定の方向への第二の所定の距離だけの移動を表す信号を生成するステップとを更に有することを特徴とする請求項第14項に記載の方法。

【請求項16】 所定の値を超過した第三の力の検出に基づいてクリック事象の発生を通知する信号を送出するステップを更に有することを特徴とする請求項第15項に記載の方法。

【請求項17】 各々入力に応答して出力を生成する複数のセンシングエレメントを有するセンサを用いる方法において、

第一の複数の前記センシングエレメントよりなるものとして第一の入力デバイスを規定するステップと、

前記第一の複数のセンシングエレメントからのデータ出力の各々を前記第一の入力デバイスの対応するデータにマッピングするステップと、

前記第一の複数のセンシングエレメントの内の少なくとも一部からのデータ出力を第二の入力デバイスの対応する出力にマッピングするステップと、を有することを特徴とするセンサ利用方法。

【請求項18】 人物を識別する方法において、人物に特徴的な力のイメージを記録するために触覚センサアレイをサンプリングするステップと、

記録された力のイメージ情報から前記人物に特徴的な情報を決定するステップと、

前記人物に特徴的な前記情報を既知の個人に特徴的な情報と比較するステップと、

前記比較段階に基づいて前記人物を既知の個人として識別するステップとを有することを特徴とする人物識別方法。

【請求項19】 人物に特徴的な前記力のイメージが手形の力のイメージを構成していることを特徴とする請求項第18項に記載の方法。

【請求項20】 前記サンプリングステップが、前記力のイメージを生成するために前記センサアレイからの連続したサンプリングによるピクセル出力の総和を計算するステップと、

各々の非零ピクセルから所定の値を減ずるステップと、0未満の値を有する全てのピクセルを0にセットするステップと、

全ての非零ピクセルと1にセットするステップとを有することを特徴とする請求項第18項に記載の方法。

【請求項21】 前記決定ステップが、

前記力のイメージの重心における原点を規定するステップと、

前記力のイメージの主軸をセットするステップと、前記力のイメージを複数の接続領域に分割するために前記力のイメージの接続領域解析を実行するステップと、

前記力のイメージの指先に対応する前記接続領域を識別するステップと、

前記原点から最も離れた各々の指先のピクセルを識別するステップと、

前記力のイメージ上の所定の点と前記原点から最も離れた各々の指先のピクセルとの間の距離を計算するステップと、

前記力のイメージの手のひらの幅を計算するステップとを更に有することを特徴とする請求項第18項に記載の方法。

【請求項22】 前記人物に特徴的な前記力のイメージが、一連の連続的な足形の力のイメージよりなることを特徴とする請求項第18項に記載の方法。

【請求項23】 前記決定ステップが、前記一連の連続的な力のイメージの第一の力のイメージの重心における原点を規定するステップと、

その後の前記連続する力のイメージの各々に対して、前記力のイメージの重心、前記力のイメージの重心と前記原点と間の距離、及び前記力のイメージの全てのピクセル値の総和、を計算するステップと、

前記人物を特徴付けるために前記各々の力のイメージを前記総和と前記重心と前記原点との間の前記距離とに関連付けるステップとを更に有することを特徴とする請求項第18項に記載の方法。

【請求項24】 前記人物を特徴付ける所定の間隔におけるデータ値を得るために前記総和及び前記距離を再補間するステップを有することを特徴とする請求項第23項に記載の方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は触覚センサアレイに関し、特にワークサーフェスを実現するために触覚センサアレイを用いる方法に関する。

【0002】

【従来の技術】多くのオフィス及び研究所には、コンピュータ及び遠距離通信システムが備え付けられている。これらのシステムは、キーボード、マウス、電話、及びファクシミリなどの、貴重なワークスペースを占有し雑然としてしまう種々の入力デバイスを必要とする。これらの種々の入力デバイスはそれが企図されたほどには便利ではない。例えば、入力デバイスをワークサーフェス上に再配置することはしばしば困難である。上述されたキーボードあるいはマウスのような多くの入力デバイスは、コンピュータシステムに対してケーブルで接続され

ており、それゆえその位置が制限されている。さらに、相異なったコンピュータユーザの便宜を図るためあるいはスペースに対する要求が変化したためにこの種のデバイスを動かすすなわち再配置することは困難であったり不便であったりする。

【0003】

【発明が解決しようとする課題】入力デバイスを整理合併する従来技術の一つは、異なった場合に異なった機能を実行するアレイを提供するものである。例えば、コンピュータのキーボードは、通常、ある場合には数字パッドキーとして定義され、またある場合には、カーソル制御キーとして定義されるキーアレイを有している。しかしながら、この種の入力デバイスは、そのアレイがまったく相異なった機能、例えばマウスとしての機能、を実現するように再配置されることを許容していない。さらに、この入力デバイスは、デバイス上のアレイの位置や配置を変更するには再配置され得ない。

【0004】他の望ましい機能、例えば安全ではないオフィスあるいは研究所に対する安価、簡潔かつ信頼するセキュリティ機能、は利用可能ではない。この種のセキュリティに対する必要性の一例は、研究所の外面上の物体の存在をモニタする能力である。例えば、研究所の化学薬品あるいは実験装置が触られた、動かされたあるいはいじられたか否かを知ることは重要である。同様に、オフィスにおいても物品が触られたあるいは動かされたか否かを知ることは重要である。

【0005】安全管理された領域あるいはコンピュータシステムへのアクセスを制限するセキュリティメジャーも改良されうる。複合鍵とコンピュータのパスワード等のセキュリティメジャーは、希望されるセキュリティのレベルを提供し得ない場合がある。例えば、複合鍵とパスワードをオーソライズされていない人物による発見等によって容易に回避され得ない他のセキュリティメジャーによって置換するあるいは増加することが望ましい。

【0006】

【課題を解決するための手段】本発明に従って、触覚センシングアレイが、情報を受容しかつ種々の入力及び通信デバイスを実現するように配置されうる多様性及び適応性を有する"インテリジェント"ワークサーフェスを生成しモニタするよう配置される。本発明に係るインテリジェントワークサーフェスは、相異なったデバイスを実現するために、望むように再配置されることが可能である。さらに、本発明に係るワークサーフェスは、その位置や配置を変更するように再配置されうる。

【0007】本発明に係るインテリジェントワークサーフェスは、インテリジェントデスクトップ、フロア、壁、あるいは他のサーフェス(表面)として実現される。望ましい実施例においては、本発明に係るインテリジェントサーフェスは、キーボード、数字キーパッド、マウスなどの入力及び通信デバイスを実現することが可

能なデスクトップサーフェスである。本発明に係るインテリジェントデスクトップは、デスクトップ上のアイテムの存在及び位置をモニタすることも可能である。インテリジェントサーフェスへの入力により、実現されたデバイスからの出力として解釈されることになる触覚アレイの出力が生成される。インテリジェントデスクトップサーフェス上に実現されたデバイスはテンプレートとして実現されることが望ましく、それゆえ貴重なワークスペースを占有したり雑然とさせたりしない。実現されたデバイスは、相異なったユーザ及びワークスペースに対する変化しつつある要求に合致するように配置及び再配置されうる。デバイスの配置をもつれさせるすなわち制限するコンピュータケーブルは存在しない。

【0008】本発明の他の実施例においては、インテリジェントワークサーフェスがフロアサーフェスとして実現される。インテリジェントフロアは、サーフェス上の物体及び人物の存在を検出しその位置を追跡することが可能である。本発明のその他の実施例においては、インテリジェントサーフェスが、人物を識別するためにサーフェスに対して印加された力の二次元イメージ("カイメージ")を生成するように配置されている。例えば、手形を生成した人物を識別するためにある人物の手形のカイメージがサンプリングされ、解析され、既知の個人のカイメージを表すデータと比較される。

【0009】

【実施例】キーボード、数字キーパッド、マウスなどの種々の入力・通信デバイスを実現する"インテリジェント"ワークサーフェスを生成するために、触覚センシングアレイが配置されてモニタされる。インテリジェントサーフェスへの入力により、触覚アレイから、実現されたデバイスからの出力としてプロセッサによって解釈されることになる出力信号が生成される。例えば、触覚アレイに印加された力により、アレイの出力において電気的信号が生成される。その後、その電気信号はプロセッサによってキーボードからのキーストローク出力として解釈される。特定のデバイスがもはや必要とされないあるいは新たなデバイスが必要とされる場合には、インテリジェントワークサーフェスは相異なったデバイスを実現するために再配置される。例えば、インテリジェントワークサーフェスは、前記具体例における触覚アレイからの出力がキーボードからのキーストロークではなくマウスからディスプレイスクリーンへの出力として解釈されるように再配置されることが可能である。

【0010】実現されたデバイスは、ワークサーフェス上で、相異なったユーザあるいは変化しつつあるワークスペースに対する要求に合致するよう、配置及び再配置される。例えば、第一のコンピュータユーザによって第一の位置にキーボードが配置される。このキーボードの位置は、例えば次のコンピュータユーザの好みに合わせるために左側にシフトさせられる。

【0011】図1は、本発明に係る触覚センサアレイサーフェス1及びプロセッサ2によって全体が覆われた机を示している。図1には、キーボード3、数字キーパッド4、入力インターフェース5、及びディスプレイスクリーン6も示されている。触覚アレイ1はプロセッサ2に対して接続されており、本発明に係るインテリジェントワークサーフェスシステムを形成している。触覚アレイ1は、プロセッサ2に対して、触覚アレイ1に対して印加された入力に関する情報を出力する。触覚アレイ1を構成しているセンシングエレメントグループ（図3を参照）は、種々のデバイスを実現する入力インターフェースとして配置されている。例えば、触覚アレイ1を構成するセンシングエレメントグループは、キーボードインターフェース3として配置される。ここで、“領域”という術語は、アレイ1上のセンシングエレメントグループを集合的に指し示すものとする。代表的なアプリケーションにおいては、キーボード3として配置された触覚アレイの領域内の所定のセンシングエレメントに印加された力は、キーストロークを表す信号を生成する。同様に、触覚アレイ1の他のセンシングエレメントグループは、例えば数字キーパッド4あるいはディスプレイスクリーン6に対する入力インターフェース5として定義される。

【0012】触覚アレイ1において実現された入力インターフェースあるいはデバイスは、デバイスの位置、大きさあるいは配置を変更する目的で望むように再配置される。さらに、新たなデバイスが触覚アレイ上に配置されて既存のデバイスが除去される。例えば、図2は、触覚アレイ1上の新しい位置を有するように再配置されたキーボード3、キーパッド4、及びディスプレイ入力デバイス5を示している。参照番号3'、4'、及び5'は、各々のデバイスの図1において示されていた位置に対応する以前の位置（破線で示されている）を示している。3'、4'、及び5'によって識別されるセンシングエレメントグループは、その元来の状態（通常未定義）にリセットされている。キーボード3と重なるセンシングエレメント3'を有する触覚アレイの部分は、キーボード3に対する適切な機能を実行するよう再定義されている。

【0013】図3は、触覚アレイ1を構成しているセンシングエレメント7を示した図である。センシングエレメント7は、電気伝導体よりなる列及び行の交点によって構成されている。触覚アレイ1に印加された入力（すなわち力）により、列及び行導体が接触する。この導体間の接触により電流が流れるようになり、列及び行導体の交点に位置するセンシングエレメントからの出力が生成される。

【0014】図3に示されているように、プロセッサ2によって数字キーパッド4として定義された触覚アレイ1の領域は、センシングエレメント7のうちの選択され

たいいくつかと重なっている。プロセッサ2は、センシングエレメントグループを入力デバイス、この場合は数字キーパッド、として定義することによって“テンプレート”機能を実行する。プロセッサ2は、センシングエレメントの出力を定義された入力デバイスの出力として解釈する“マッピング”機能も実行する。例えば、キーパッド4のキー“1”及び“4”は、それぞれ参照番号8及び9によって集合的に示されているセンシングエレメントグループと重なっている。プロセッサ2は、センシングエレメント8及び9からの（これらのエレメントに対して印加された力の結果としての）出力をキーパッド4の出力として解釈する。

【0015】数字キーパッド4は、プロセッサ2内でキーパッドに対応する触覚アレイ1の領域の定義を変更することにより再配置される。この実施例においては、領域の再定義は、キーパッド4のキーとして機能する相異なったセンシングエレメント7を規定することにより実現される。例えば、キーパッド4はプロセッサ2によってキーパッドのキー“1”が（図3において示されているように参照番号8によって指し示されているセンシングエレメントグループではなく）参照番号9によって指し示されているセンシングアレイグループと重なるように再定義される。すなわち、キーパッド4が下方にシフトされ、“1”キーが“4”キーによって占められていた位置を占め、“2”キーが“5”キーによって占められていた位置を占める等。再定義されると、プロセッサ2はセンシングエレメント9からの出力を、最初に定義されていたように“4”出力ではなく、キーパッド4からの“1”出力として解釈する。

【0016】図4は、本発明に係るインテリジェントワークサーフェス（すなわち、図1の触覚アレイ）あるいはそのようなワークサーフェスを複数個実現するために用いられる並列に接続された複数個の触覚センサアレイ12を示した図である。アドレス回路14が各々のセンサアレイ12に関連させられており、各々のアレイ12を構成している（“ピクセル”とか“タクセル”と呼称される）個別のセンシングエレメントをアドレッシングする。回路14は、アプリケーションスペシフィック集積回路すなわち“ASIC”として示されている。（サンプルホールド機能を有する）アナログ-デジタル（A/D）変換回路16がアレイ12からアドレス回路14を介してデータ出力を受信する。（アレイの特性によって決定される）所定のセトリングタイムの後、データはA/Dコンバータからデータバス20を介してアドレス可能な3ステートバッファ18に渡される。データバス20は、本発明の原理に従ってデータを処理する図1及び図2のプロセッサ2などの適切なプロセッサを接続している。図4のセンサアレイ回路の具体的な実現の詳細に関しては、以下においてインテリジェントワークサーフェスの実現に関する具体的な技法に係る議論の後に議

論される。

【0017】図5から図7は、プロセッサ2において種々のデスクトップアプリケーションを実現するための具体的な方法を示した流れ図である。図11及び図12は、プロセッサ2などのプロセッサにおいて種々のインテリジェントフロアアプリケーションを実現するための具体的な方法を示した流れ図である。

【0018】本発明に係る種々のアプリケーションの詳細を記述する前に、ほとんどのアプリケーションに対して共通の2つの画像処理技法が記述される。これらの技法は、“連結領域解析”(すなわち“クラスタリング”)であり、画像の重心を決定するものである。クラスタリングは、12個のアレイよりなるサンプルピクセルを連結領域にグループ分けする段階を有している。クラスタリング技法は画像処理の分野において公知であり、ティー・パブリデス(T. Pavlides)による“グラフィックス及び画像処理に関するアルゴリズム”(コンピュータ・サイエンス・プレス社、1982年)に記載されている。クラスタ化された領域の各々の重心は、n×m個のセンサアレイに関して以下に示される数式を用いて決定される：

【数1】

$$\bar{x} = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m i * V_{ij}$$

【数2】

$$\bar{y} = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m j * V_{ij}$$

ここで、n及びmは各々センサアレイの列数及び行数、i及びjは各々列及び行の変位、Vは座標(i, j)のピクセルのグレイスケール値である。

【0019】デスクトップアプリケーションにおいては、データはプロセッサに対して以下の3種類の型のインターフェースの内の一つを介して入力される：“キーボード”インターフェース、“マウス”インターフェース、及び“画像”インターフェース。デスクトップサーフェスの各部分は、初期化手続きにおいて、これらの型の内の一つのインターフェースとしての入力デバイスとして規定される。プロセッサは入力デバイスを前述されたテンプレート及びマッピング機能によって規定する。すなわち、プロセッサは、まず、選択されたセンシングエレメントを入力デバイスとして識別する。その後、プロセッサは選択されたセンシングエレメントの出力を入力デバイスの出力として解釈する。プロセッサは、デスクトップ上の相異なった部分を、複数個の入力デバイスとして同時に規定しうる。

【0020】図5の流れ図は、本発明に従ってキーボードインターフェースを実現する具体的な方法を示した図である。キーボードインターフェースは、(図1に示さ

れたキーボード3のような)キーボード、数字キーボード、及び他の同様のデバイスを提供するために用いられる。(これらのデバイスは、柔らかい材料よりなるシートをパターンが形成されたテンプレートと重ね合わせることによって構成される。)以下に詳細に記述されているように、図5に示された方法は、他の、キーボードではないデバイスを規定するためにも用いられる。

【0021】ボックス30に示されているように、プロセッサ(例えば図1のプロセッサ2)はセンサアレイ12をサンプリングする。このことは、例えば各々のセンシングエレメントをアドレッシングするためにアレイ全体にわたってサイクルすることによって実行される。その後、センシングポイントは連結領域に“クラスタリング”され、プロセッサはそれぞれの領域の重心を計算する。データがアレイから集められると、プロセッサは3つの一般的な操作の内の一つを実行する。すなわち、プロセッサはデバイスを規定するか、既に規定されたデバイスに対する処理操作を実行するか、あるいはある所定の事象(例えば、規定された領域外の触覚アレイ1への接触)の発生による“オプション”の、すなわち特別の処理操作を実行するかのいずれかである。

【0022】ボックス32、34、36、38は、モジュール(すなわち、センサアレイ上のキーボードやマウスなどと規定された領域)が“アクティブ”なデバイスのリストに追加される方法を示している。アレイは、少なくとも一つのクラスタが検出されるまでサンプリングされる。クラスタを検出すると、プロセッサは、3つのクラスタが存在するか否かを決定する。3つのクラスタが検出された場合には、プロセッサはその3つのクラスタの重心が既知のモジュールの“足跡”に対応しているか否かを決定する。その場合には、そのモジュール及びその位置及びその方向がアクティブデバイスのリストに追加される。重心が既知のモジュールの“足跡”に対応しない場合は、入力が無視され、再びアレイがサンプリングされる。

【0023】ひとたびデバイスがアクティブとなると、プロセッサは、例えばアクティブ領域内のキーが押されたか否かを決定するためにアレイをモニタする。決定ボックス34への応答が“ノー”である(すなわち、クラスタ数が3に等しくない)場合は、プロセッサは一つのクラスタのみが存在するか否か(ボックス40)及びこのクラスタがどのデバイスに属しているのか(ボックス44)を決定する。各々のデバイスが重なりを有さない領域に分割されるため、単一のクラスタの重心は、例えばキーボード上の押された“キー”を識別する単一の領域内に存在する。プロセッサ2上で実行されているデバイススペシフィックなソフトウェアは、触覚アレイの出力を規定された入力デバイスの出力として解釈し、検出されたキーストロークに基づいて所定の機能を実行する。

10

20

30

40

50

【0024】ボックス40における決定結果が“ノー”である(すなわち複数個のクラスタが存在する)場合には、プロセッサはセンサアレイ上への複数個の接触に基づいて機能する付加的な処理を実行することが望ましい(ボックス48)。このことにより、システムのオン/オフ、最後の入力削除、セッションの終了あるいは再起動、あるいは他の型のモジュール(例えばマウス)に対する領域規定などのモジュールに依存しない機能を許容する手段が供給される。同様に、決定ボックス42への答えが“イエス”である場合には、プロセッサはあらゆる既知のモジュールのアクティブ領域外への接触に対する付加的な処理(例えば、計算機領域外を押すことにより計算のための直前のキーストロークを削除するなど)を実行することが望ましい。

【0025】触覚アレイ1上で第二の型のインターフェース、すなわちマウスデバイス、を実現するための方法の具体例が図6の流れ図に示されている。まず、触覚アレイ1上でマウスのアクティブ領域(すなわち、そのデバイスの範囲を規定するセンシングエレメント)が規定されなければならない(ボックス50)。このことは、図5の流れ図に示されたステップを用いてなされる。その後、マウスは3つの相異なった操作を実行することによって実現される。第一に、マウス領域の原点が設定されなければならない。この原点はディスプレイモニタのスクリーン上のカーソルの位置を設定する。第二に、クリックという事象が検出されて表示されなければならない。クリック事象は、所定の処理操作がなされなければならないことを表すマウスデバイスへの入力である。第三に、ディスプレイモニタのスクリーン上のカーソルが、マウス領域の原点の再配置にตอบสนองして再配置されなければならない。

【0026】マウスのアクティブ領域が決定されると、規定されたマウス領域内でクラスタが検出されるまで触覚アレイがサンプリングされる(ボックス52)。クラスタが検出されると、プロセッサは原点が設定されたか否かを決定する(ボックス56)。原点が設定されていない場合には、処理は、図示されているように、ボックス58へ進む。プロセッサはセンサアレイのマウス領域を数回サンプリングする。重心の平均が、重心のx及びyの値を仮総和に追加し、その仮総和をなされたサンプリングの回数で割ることにより計算される。その後、原点がマークされる。所定の回数のサンプリングがなされたことを保証するためにサンプリングカウンタが用いられる(ボックス60)。サンプリング回数は、そもそもの接触の原点を十分な信頼性を有して配置するために必要とされる回数を超過していなければならない。元となるクラスタがアクティブマウス領域内に存在しない場合には、仮総和はリセットされ(ボックス54)、原点は設定されず、新たなサンプリングがなされることになる。

【0027】前述されているように、ひとたび原点が設定されると、プロセッサはアレイのサンプリングを続行する。プロセッサは、続行されたサンプリングに対して、各々のサンプリングクラスタ内のピクセル値の総和が“クリックスレッシュールド”を超過したか否かを決定する。クリックスレッシュールドは、それを超過するとマウスの“クリック”事象が発生したと仮定されるような所定の値である。クリックスレッシュールドを超過することは、従来のマウスのボタンを押し下げたことと同一の効果をもっている。クリック事象が発生すると、プロセッサはクリック事象の発生に係る特定のアプリケーションによって要求されるインストラクションを実行する(ボックス66)。クラスタのピクセル値の総和がクリックスレッシュールドを超過しない場合には、プロセッサは、クラスタの重心が原点の重心から所定の距離(“原点距離”)だけ移動したか否かを決定する。移動していない場合には、触覚アレイが再びセンサされる。重心が十分な距離だけ移動した場合には、関連するディスプレイモニタのスクリーン上に表示されているカーソルが新たなサンプリングがなされる前に新しい位置に移動させられる(ボックス70)。ディスプレイモニタ上のカーソルは、元の位置から現時点での重心の位置方向に移動させられる。カーソルは、“速度定数”と原点と現時点での重心との距離との積に等しい距離だけ移動させられる。速度定数は、センサの解像度、ディスプレイモニタの解像度、及びサンプリングレートに依存し、如何に速くカーソルが移動すべきかというユーザの好みに合わせるようユーザによって変更されることが望ましい。新たな重心にตอบสนองしてカーソルを移動するための技法は、従来のマウスの動きにตอบสนองしてディスプレイカーソルを移動させる技法と同様である。これらの技法は当業者には公知である。

【0028】画像インターフェースが図7を参照して記述される。画像を処理する場合には、センサアレイがサンプリングされ(ボックス72)、そのアレイからのグレイスケールピクセル値(あるいは他のマルチスケール値)が画像(すなわち、各々のピクセルが列及び行アドレスを有している)としてストアされる(ボックス74)。その後、画像グレイスケールデータは特定のアプリケーションに対して必要とされるように処理される(ボックス76)。その種のアプリケーションの一例が、図8から図10を参照して記述される。

【0029】図8は、個人をその手形に基づいて識別する具体的な方法を示している。このアプリケーションは、例えば管理領域あるいはコンピュータ端末への個人のアクセスを保証あるいは否定するために用いられる。このプロセスの第一段階は、ボックス78に示されているように、連続したアレイのサンプリングを行ない、かつそれらのサンプルデータの総和を計算して手形の全体像を形成することである。プロセッサはセンシ

10

20

30

40

50

グレイを例えば50Hzでサンプリングし、信頼しうる画像を得ようサンプリングデータの総和を求める。

【0030】その後、当該画像に対して、全ての非零ピクセルの値を見出すことによってスレッシュールドがかけられ、各々の非零ピクセルから所定の値（例えば平均のピクセル値に0.3を乗じたもの）が差し引かれる。この減算に引き続いて、0未満の値を有する全てのピクセルを0に設定することにより、さらにスレッシュールドがかけられる。このプロセスにより雑音除去され、プロセッサによるクラスタリングが行なわれるあらゆる時間の間に、人間がセンシングアレイに対して手を保持しておくことを許容する。

【0031】次のステップ（ボックス82）は、グレイスケール手形画像を2値画像に変換することである。非零ピクセルに対しては、1の値が与えられる。このことにより、各々のピクセルに対して同一の重みが与えられたことになり、その後の操作に対して、グレイスケールの重心を用いて利用可能なものよりもより良い基準が提供される。図9は、このプロセスによって得られた手形画像106を示した図である。この処理方法の残りの部分は図9及び図10を参照して記述される。

【0032】次のステップ（図8のボックス84）は、2値画像106の重心108と主軸を配置することである。重心108は、前述されたクラスタリングに関する式を用いて決定される。ここで、全ての非零ピクセル値は1である。原点を重心108に有する象限のX軸110はより小さい主モーメントに対応する主軸に平行であり、手のひらの付け根の部分に向かっている。Y軸112はX軸110に垂直であり、右手座標系を構成している。このことにより、手の向きが規定される。

【0033】2値画像は連結領域にクラスタリングされ（ボックス86）、各々の連結領域の重心が決定される。その後、親指の先に対応するクラスタ114が配置される（ボックス88）。クラスタ114はX軸110から最も離れた重心を有するクラスタであり、その重心のx座標は-1.2から3.5インチの間の値を有する。

【0034】次に、右手であるか左手であるかが決定される。親指の重心のy座標が正であればその手形は左手であり、y座標が負であれば右手である。

【0035】次に、各々の指先に対応するクラスタが識別されて順序付けられる（ボックス90及び92）。このことを行なうために、プロセッサは重心108と負のx座標を取る重心を有する各々のクラスタ（親指は除く）の重心との間の距離を計算する。プロセッサはこの値を重心のx座標の2乗に加算する。最大の値を有する4つの重心が指先である。その後、指先はそのy座標に従って親指を含めて順序付けられる。手形が左手である場合は、親指は最大のy座標を有しており、人差し指はその次、さらに中指、薬指、小指と続く。手形が右手で

ある場合はその逆となる。

【0036】ボックス94に示されているように、プロセッサは、その後、指先に対応する各々のクラスタ114、116、118、120及び122の重心から最も遠いピクセルの位置を決定する。

【0037】次に、手のひらの付け根の中心点124の位置が決定される（ボックス96）。このことを行なうために、図10に示されている新たな座標軸が決定される。この新たな座標軸は、もとの座標軸と同一の原点に基づいている。新たなX軸は、中指に対応するクラスタ120内の重心から最も遠いピクセル126と、親指に対応するクラスタ114の重心と小指に対応するクラスタ116の重心とを結んだ線分の中点128とを結ぶ直線に平行であるように決定される。新たなY軸134により右手座標系が完成する。この座標軸を用いると、手のひらの付け根の中心点124のx座標は、新たな座標軸における最大のx座標を有する非零ピクセルのx座標となる。y座標は、x座標が0より大きく、それぞれ最大及び最小のy座標を有する2つの非零ピクセル（親指に対応するピクセル114は除く）を見出すことによって決定される。この2つのy座標の平均が、手のひらの付け根の中心点124のy座標となる。

【0038】その後、プロセッサは手のひらの幅を決定する（ボックス98）。手のひらの幅は、指として識別されたクラスタを除いたy座標の最大値と最小値との間隔の絶対値として定義される。プロセッサは、さらに、手のひらの付け根の中心点124と各々の指先の最も遠いピクセルとの間の距離も決定する（ボックス100）。

【0039】個人の手形を特徴付けるために、6つの数値からなる組が用いられる（ボックス102）。この6つの数値の組を構成しているのは、手のひらの付け根の中心点124から各々の指先の最も遠いピクセルまでの距離、及び手のひらの幅である。6つの数値の組は、手形のライブラリを構成している6つの数字の組のライブラリと比較される（ボックス104）。よって、未知の個人の手形が6つの数値の組 d_1, \dots, d_6 を計算し、それを既知の個人に対してストアされた6つの数値の組 c_{x1}, \dots, c_{x6} と比較することによって識別される。この比較は、次式に従って、既知の各々の個人に対する6つの数値の組との間の距離を計算することによってなされる：

【数3】

$$D_x = \left(\sum_{i=1}^6 (c_{x,i} - d_i)^2 \right)^{1/2}$$

未知の手形は、 D_x の最小値に係る既知の個人に属するものと識別される。

【0040】図11及び図12の流れ図は、本発明に従ったインテリジェントフロアアプリケーションをプロセ

10

20

30

40

50

ッサ2において実現するための具体的方法を示した図である。

【0041】図11は、センサアレイによってカバーされている（フロアなどの）表面上で動作させられているモーター付き台車を認識しかつ制御するための方法を示した流れ図である。台車は識別され、その位置が決定される。その後、台車はフィードバック制御下で希望する位置まで移動させられる。台車と他の物体との衝突は、プライオリティ（優先順位）を決定する手続きを用いて回避される。

【0042】モーター付き台車を制御する第一のステップは、ボックス140に示されているようにフロアセンサアレイをサンプリングすることである。その後、このプロセスは、台車をその“足形”で識別する（ボックス142）。このことを行なうために、プロセッサは前述されているようなクラスタ解析を行ない、検出されたクラスタの重心及び重心間のすべての距離を決定し、この情報を台車に関する既知の足形と比較する。その後、プロセッサはフロア上の台車の位置を決定し（ボックス144）、その位置を希望する位置と比較する。利用可能である場合には、台車の直前の位置が台車の現時点での位置の検索を制限するために用いられる。台車のモーター及び舵取りユニットが（動作させられている台車の型に対して要求されているように）起動させられて台車が希望する位置へ駆動される。プロセッサはフロア上のセンシングアレイを繰り返しサンプリングし、台車が希望する位置に到達するまでモーター及び舵取りユニットを調節する（ボックス146）。

【0043】プロセッサは、例えば他の台車や人間などの障害物との衝突を回避する手段を提供することが望ましい（ボックス141）。ある実施例においては、台車が他の台車（あるいは物体）から所定の距離内に入ると、2つの台車間の相対的な優先準位が決定される（ボックス143）。より高い優先準位を有する台車が希望する位置まで駆動される（ボックス145）。低い優先準位を有する台車は、高い優先準位を有する台車が範囲外に去るまで停止されたままとなり（ボックス147）、その後希望する位置まで駆動される。台車には、通常、動かない物体よりも高い優先準位が与えられるが、人間よりは低い。台車の方向及び速さは、障害物を回避するために必要となるように、及びフロア上の交通パターンに依存して修正される。

【0044】図12は、“足形”解析に基づいて個人を識別する方法を示した流れ図である。足形解析の第一ステップは、ボックス148に示されているように、アレイをサンプリングすることである。足跡が一定のレートでサンプリングされなければならない、その結果一連の足形画像よりなる足跡のテンプレートが構成されることになる。一度構成されると、その“未知の”（すなわち未だ識別されていない）足跡のテンプレートが“既知の”

足跡のテンプレートのライブラリと比較され、未知の足跡が識別される。

【0045】プロセッサは、足跡のかかと部分がフロアに接触したのを検出したセンサアレイサンプルを第一フレーム（足跡の第一の足形画像）として規定することによって足跡の“原点”を計算することが望ましい（ボックス150）。この第一の足形画像の重心及び第二の足形画像が計算されてその平均が計算される。この重心の平均が原点を規定する。

10 【0046】その後の足形画像の各々に対して、足跡の端において爪先がフロアから離れるまで、プロセッサはセンサアレイをサンプリングし、足形画像の重心を計算しそれをストアする（ボックス152、154、及び156）。プロセッサは、さらに、各々の足形の重心から原点までの距離（ボックス158）、及び全てのピクセル値の総和（ボックス160）を計算してストアする。全てのピクセル値の総和は、アレイに印加された重量に比例する。全ての足形サンプルが取られると（すなわち、足がもはやフロアと接触していなくなると）、プロセッサはデータを再び補間する（ボックス162）。

20 【0047】一組の足跡においてプロセッサがN個のデータ（すなわち、N個の足形画像）をサンプリングすると、(N-2)個の計算された距離及びピクセル値の総和（すなわち重量）が得られることになる。図13に示されているように、この方法においては、距離と重量との関係に注目している。図13のグラフは、かかとがセンサアレイに接触した時刻から爪先がセンサアレイから離れた時刻までの間の一連のアレイのサンプリングを通じて集められた時間と重量及び時間と距離の情報から導出されたものである。横座標上で等間隔のデータを得るために、データが15/128距離単位の間隔で再補間されることが望ましい。このことにより、0から15距離単位（たとえばインチ）の間に128個のデータポイントが得られる。（15という数は、足形が最大でも15インチの長さを超過しないことから選択された。）これらの128個のデータ（そのうちの幾つかは0である可能性がある）は、未知の足跡を特徴付ける128個の数よりなる組 $\langle d_1, \dots, d_{128} \rangle$ を規定する（図12のボックス164を参照）。未知の足跡に係る128個のデータの組は、足跡のライブラリ中の既知の個人に対する128個のデータの組（個人“x”）に対する $\langle C_{x1}, \dots, C_{x128} \rangle$ と比較される（図12、ボックス166）。足跡は、ユークリッド幾何学的な距離測定【数4】

$$D_x = \left(\sum_{i=1}^{128} (d_i - C_{x,i})^2 \right)^{1/2}$$

を用いて比較される。未知の足跡（及びその未知の足跡が属する未知の個人）は、最小の D_x を有する足跡（及び個人）として識別される。

【0048】図4に戻って、本発明に係るインテリジェントワークスペースを実現する具体的な回路の詳細が以下に記述される。センサアレイは、表面の高さあるいはプロファイルを著しく変更することなくワークサーフェスをカバーすることが望ましい。センサアレイ12の各々は、通常、ポリマーインクアレイなどのピエゾ抵抗触覚センサアレイである。センサアレイ12は個別のアドレス可能なセンシング素子を有しており、これらは通常列導体及び行導体の交点を形成している。よって、アレイ12は行及び列を順次アドレッシングすることにより”サンプリング”される。アレイ12用に適したセンサアレイは、インターリンクエレクトロニクス (Interlink Electronics) 社 (カリフォルニア州カービンテリア (Carpinteria)、93013) より市販されている。もちろん、例えば可変容量センサなどの他のタイプのセンサが本発明の範疇及びその精神を逸脱することなく用いられうることは当業者には自明である。

【0049】アドレッシング回路14は、通常、アレイ12におけるパラレルバスを望ましくない電流が流れるのを防ぐために、各々のセンシング素子をアイソレートする回路を有している。アドレッシング回路14を実現するための回路は、本発明の発明者が共発明者となっている出願中の米国特許出願第07/775、881号 (1991年10月11日付け) に記載されている。A/Dコンバータ16は、市販のA/Dコンバータ回路を用いて実現されうる。

【0050】アドレッシング回路14、A/Dコンバータ16、及びバッファ18はコントロール線22を介してコントロールされる。コントロール信号は回線22を介して全てのセンサアレイ12に対して同時に伝達され、データはバッファ18を介して共通データバス20上にシケンシャルに書き出される。ストロープ信号により、センサアレイ12からの (アドレッシング回路14を介した) アナログデータ出力のA/Dコンバータ16による変換が行なわれる。

【0051】アレイを選択するため及びデータをアレイからバス20に出力するために必要とされる時間がセトリングタイム未満である限り、センサアレイ12のすべて (センサ選択コードの限界まで) が単一のアレイとして同一レートでアクセスされうる。2ⁿ個より多いセンサアレイ12が要求される場合には、付加されたセンサアレイを適応させるためにデータバスが拡張されうる。

【0052】インテリジェントワークサーフェスアプリケーションにおいて決定さるべき一時パラメータは、センシングエレメントの密度及びアレイのサンプリング周波数である。これらのパラメータは、アプリケーションによってほぼ制御される。例えば、インテリジェントデスクトップは、一般に、インテリジェントフロアよりも大きいセンシングエレメント密度を要求する。

【0053】32 μ sのセトリングタイム、800nsのストロープ (市販の代表的なA/Dコンバータのように、ストロープ後800nsした時点でデータが確定するものとする) 及び1 μ sのアドレッシングピリオドを仮定すると、アレイのセンシングエレメントは平均して33.8 μ s毎にアドレスされてサンプリングされる。よって、16 \times 16個のセンサよりなるアレイ (すなわち、各々のアレイは256個のセンシングエレメントを有している) は、7.54Mbits/secのレートでデータを生成する。このデータレートは、30Hzで更新されて7.37Mbits/secのデータを生成する512 \times 480ピクセルよりなる標準的なビデオ画像と同様のものである。

【0054】代表的なモジュラー入力デバイス、例えばキーボードやマウス、を実現するためには、0.125から0.25平方インチのセンシングエレメント密度が通常適切である。この分解能では、インテリジェントデスクトップは1.2から4.9Mpixels/secのレートでデータを生成する。これは、30Hzのビデオ画像と同程度のものである。

【0055】12フィート \times 12フィートが通常のオフィスあるいはオフィスビルの小さな共用装置領域である。このような大きさの領域において、1.5から0.25平方インチのセンシングエレメント密度を用いると、0.27Mpixels/secから9.78Mpixels/secのレートでデータが生成されることになる。中間の値としてセンサ密度3/4平方インチを用いると、インテリジェントフロアは1.10Mpixels/secのレートでデータを生成する。これは、単一のビデオカメラによって生成されるデータのおよそ15%である。これらのデータレートは、例えば前掲の米国特許出願第07/775、881号に記載されているように、サンプリングレートを115Hzまで上昇させることあるいはアレイの分解能を上昇させることによって実質的に変化させられうる。

【0056】以上の説明は、本発明の一実施例に関するもので、この技術分野の当業者であれば、本発明の種々の変形例が考え得るが、それらはいずれも本発明の技術的範囲に包含される。

【0057】

【発明の効果】以上述べたごとく、本発明によれば、多様性及び適応性を有するインテリジェントワークサーフェスが提供される。

【図面の簡単な説明】

【図1】本発明のインテリジェントワークサーフェスとして配置されたセンシングアレイによって覆われたデスクトップサーフェスを模式的に示す図である。

【図2】本発明のインテリジェントワークサーフェスとして配置されたセンシングアレイによって覆われたデスクトップサーフェスを模式的に示した図である。

【図3】図2に示されたインテリジェントワークサーフェスの一部を分解した様子を簡潔に示した図。

【図4】本発明の原理に従うインテリジェントワークサーフェスを実現する回路を示したブロック図。

【図5】本発明のインテリジェントデスクトップサーフェス上でキーボードタイプのインターフェースを実現するための流れ図。

【図6】本発明のインテリジェントデスクトップサーフェス上でマウスタイプのインターフェースを実現するための流れ図。

【図7】本発明のインテリジェントデスクトップサーフェス上で画像タイプのインターフェースを実現するための流れ図。

【図8】本発明のインテリジェントデスクトップサーフェスを用いて手形によって個人を識別するための流れ図。

【図9】本発明のインテリジェントワークサーフェス上で生成された手形画像を示す図。

【図10】本発明のインテリジェントワークサーフェス上で生成された手形画像を示す図。

【図11】本発明のインテリジェントフロアサーフェスを用いて可動式機械を制御するための流れ図。

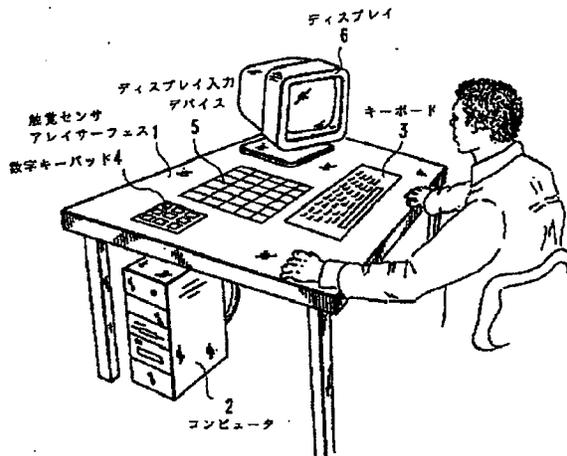
【図12】本発明のインテリジェントフロアサーフェスを用いて足形解析から個人を識別するための流れ図。

【図13】連続する足形画像サンプルから得られたデータを示すグラフ。

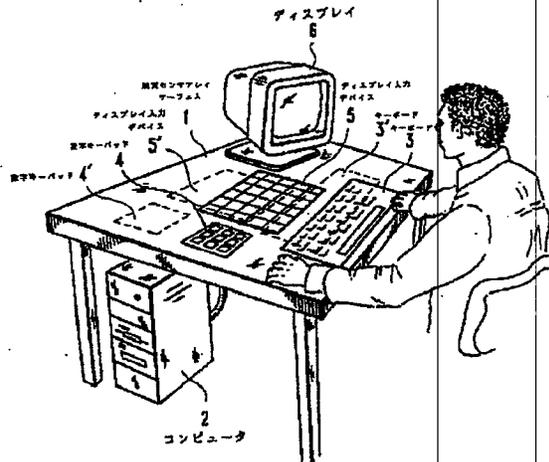
【符号の説明】

- * 1、1' 触覚センサアレイサーフェス
- 2 コンピュータ
- 3、3' キーボード
- 4、4' 数字キーパッド
- 5、5' ディスプレイ入力デバイス
- 6、6' ディスプレイ
- 8、9 センシングエレメントグループ
- 12 センサアレイ
- 14 アドレス回路
- 10 16 A/Dコンバータ
- 18 3ステートバッファ
- 20 データバス
- 22 制御回線
- 106 2値画像
- 108 重心
- 110 X軸
- 112 Y軸
- 114 親指の指先
- 116、118、120、122 指先
- 20 124 手のひらの付け根の中心点
- 126 重心から最も遠いピクセル
- 128 クラスタ114の重心130とクラスタ116の重心132とを結ぶ線分の中点
- 130 クラスタ114の重心
- 132 クラスタ116の重心
- 134 新たなY軸
- * 136 新たなX軸

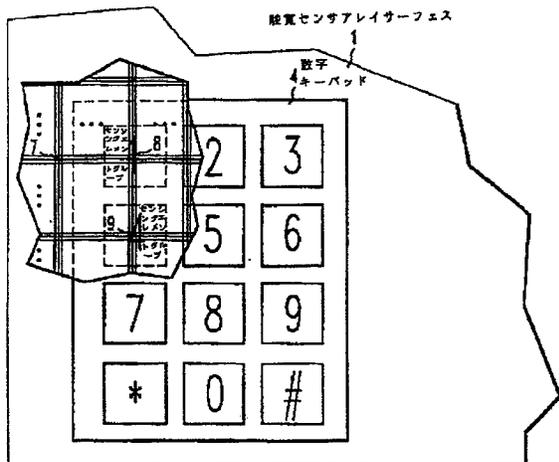
【図1】



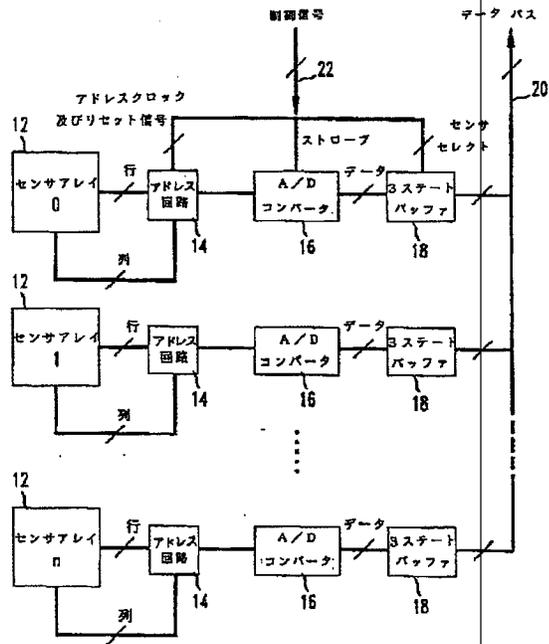
【図2】



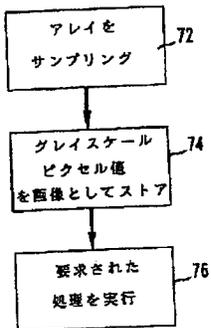
【図3】



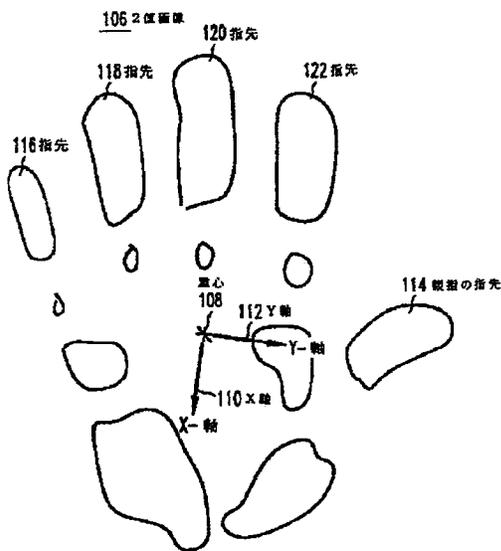
【図4】



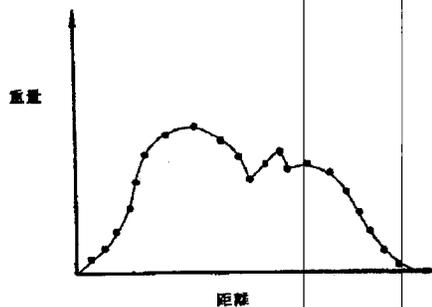
【図7】



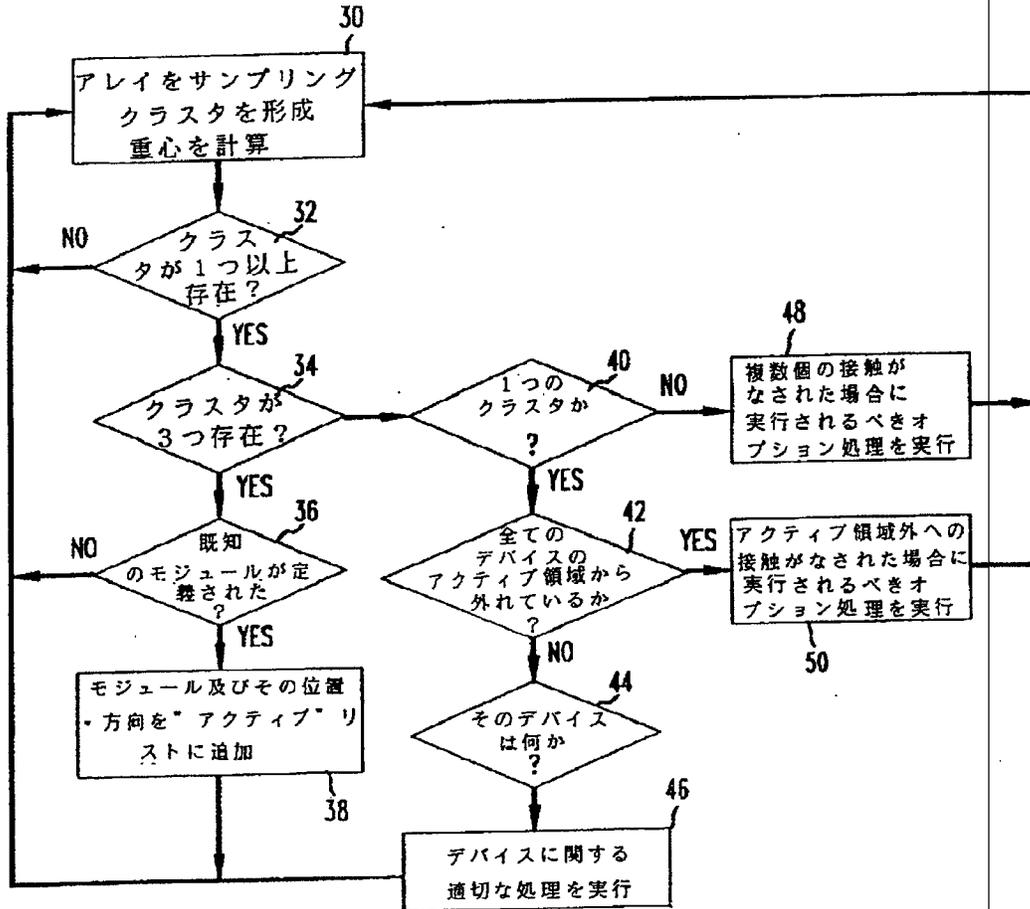
【図9】



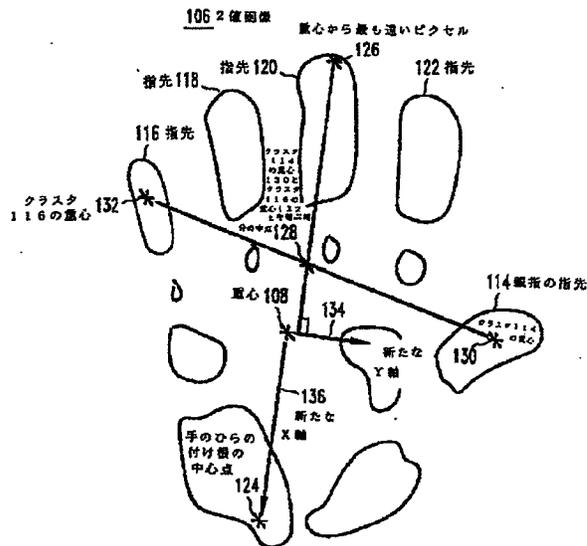
【図13】



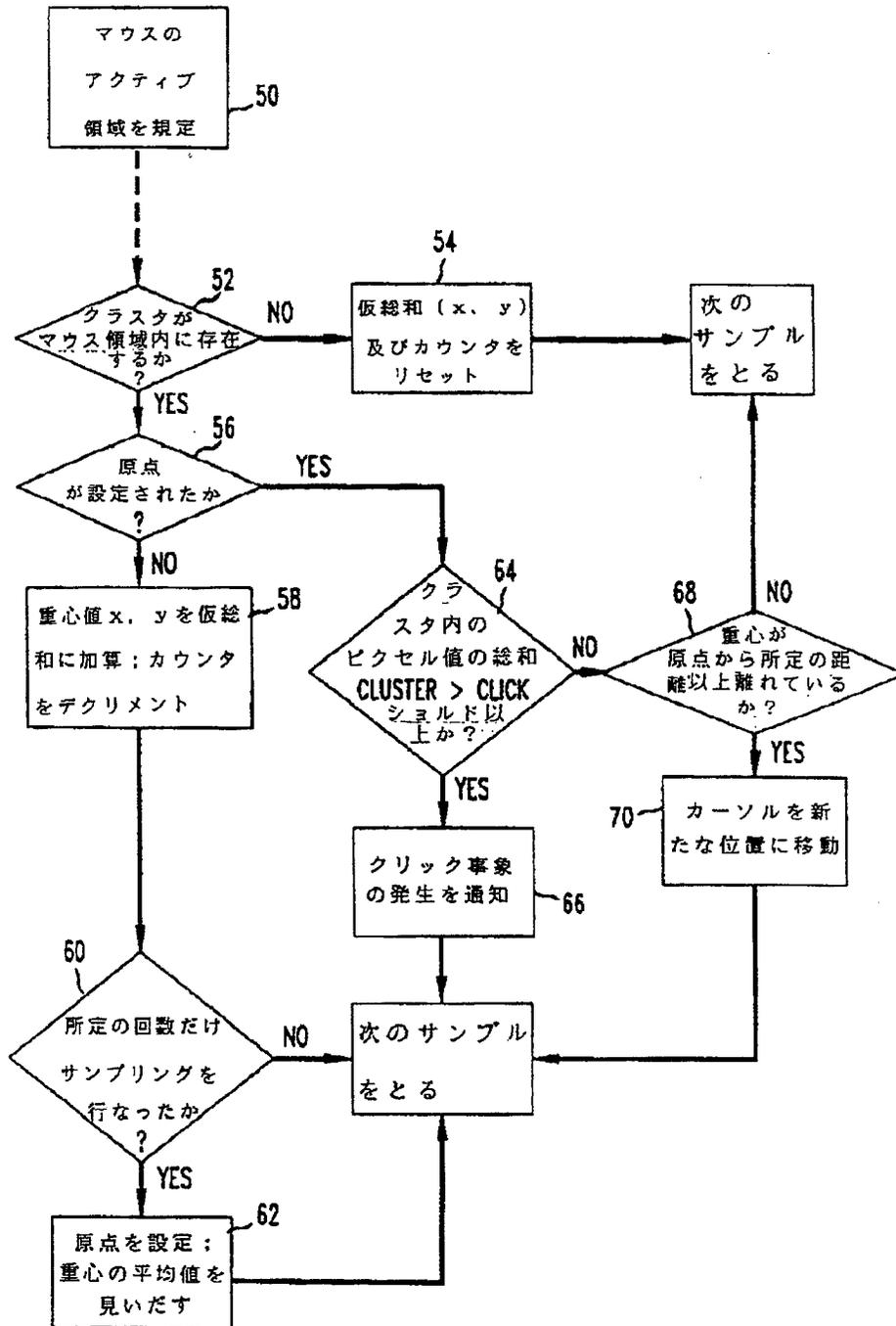
【図5】



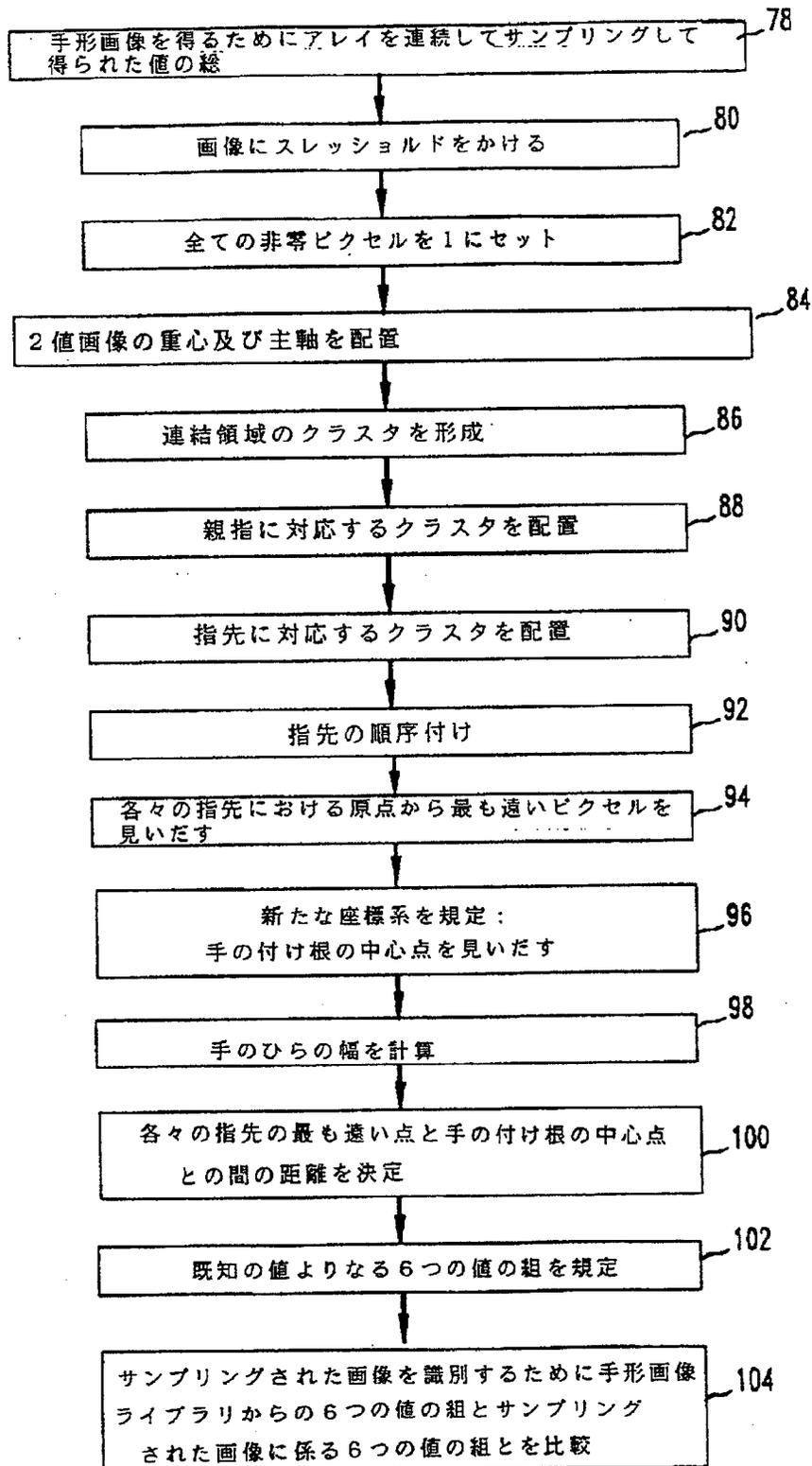
【図10】



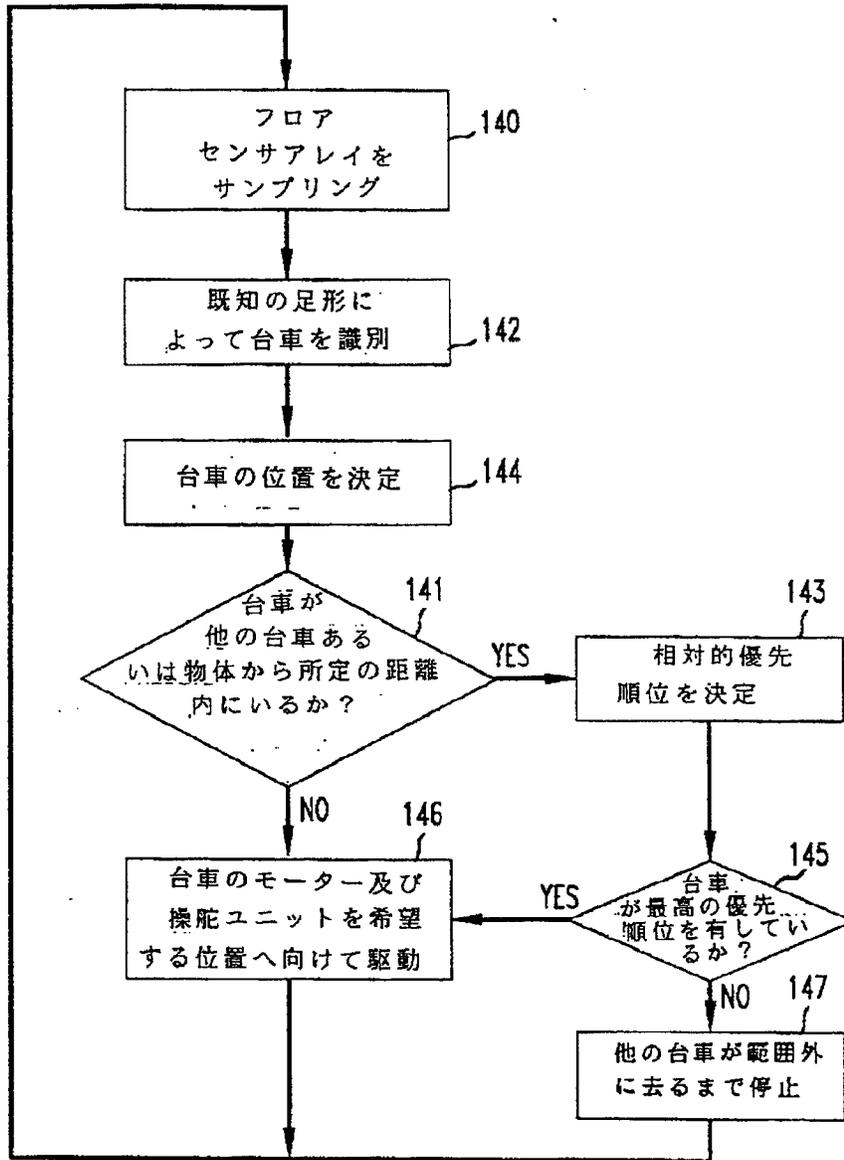
【図6】



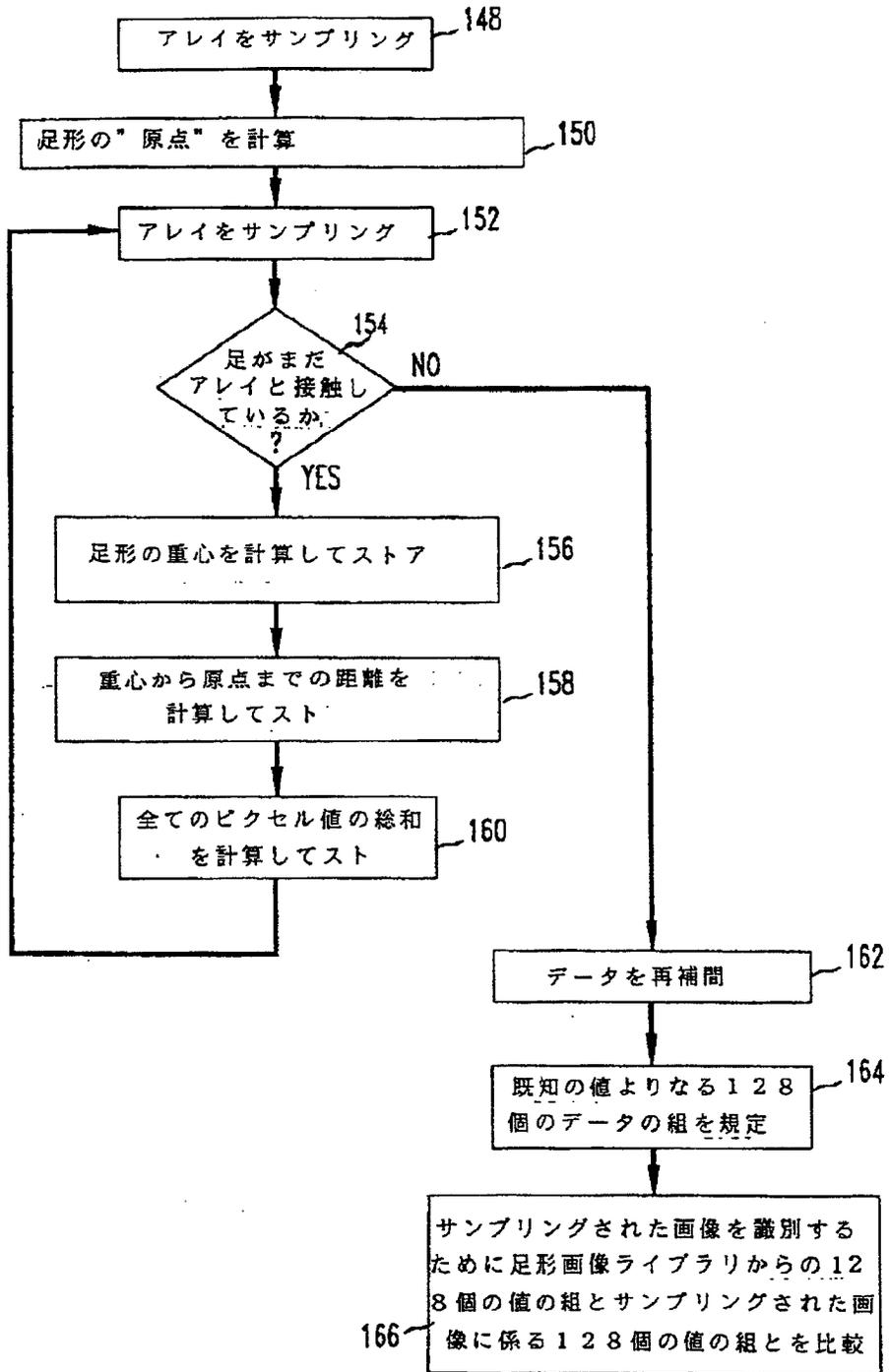
【図8】



【図11】



【図12】



19



Europäisches Patentamt
European Patent Office
Office européen des brevets



11 Publication number: **0 546 704 A2**

12

EUROPEAN PATENT APPLICATION

21 Application number: **92310595.1**

51 Int. Cl.⁵: **G06F 3/033, G07C 9/00,
A61B 5/117**

22 Date of filing: **20.11.92**

A request for correction of the numbering of claims number 29 as to read number 28 has been filed pursuant to Rule 88 EPC. A decision on the request will be taken during the proceedings before the Examining Division (Guidelines for Examination in the EPO, A-V, 2.2).

71 Applicant: **AMERICAN TELEPHONE AND TELEGRAPH COMPANY**
32 Avenue of the Americas
New York, NY 10013-2412(US)

30 Priority: **13.12.91 US 807524**

72 Inventor: **Speeter, Thomas H.**
429 Ely Harmony Road
Freehold, New Jersey 07728(US)

43 Date of publication of application:
16.06.93 Bulletin 93/24

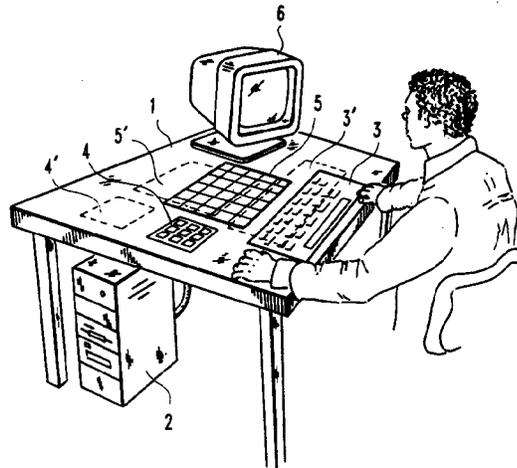
74 Representative: **Johnston, Kenneth Graham et al**
AT&T (UK) Ltd. 5 Morningson Road
Woodford Green Essex, IG8 OTU (GB)

84 Designated Contracting States:
DE FR GB IT SE

54 **Intelligent work surfaces.**

57 Methods and apparatus for implementing intelligent work surfaces are provided. Work surfaces, such as desktops and floors, are covered with a tactile sensing array to provide tactile data to a processor. Regions on the sensing arrays may be defined to include predetermined sensing elements of the sensing array and may be specified to function as input devices, such as a keyboard, a mouse, or the like. Once defined, regions may be redefined to reposition a specified input device. Regions also may be respecified to function as a different input device. The arrays also may collect data at predetermined intervals for analysis, such as for identifying an individual or an object. For example, the force-image of an individual's handprint or footprint could be taken, analyzed, and compared against data representing the force-images of known individuals.

FIG. 2



EP 0 546 704 A2

Field of the Invention

The present invention relates to tactile sensor arrays and, more particularly, to methods for using tactile sensor arrays to implement work surfaces.

Background of the Invention

Many offices and laboratories are equipped with computer and telecommunications systems. These systems require a variety of input devices, such as keyboards, mice, telephones, and facsimile machines, which occupy and clutter valuable workspace. The various input devices are not as convenient as they could be. For example, it often is difficult to rearrange input devices on a work surface. Many input devices, such as the keyboard and mice described above, are connected to a computer system by cables and therefore are restricted in position. It also may be difficult or inconvenient to move or reorient such devices to accommodate different computer users or changing space requirements.

One prior art technique for consolidating input devices was to provide an array that performed different functions at different times. For example, a computer keyboard typically includes an array of keys that may be defined as a numeric keypad at one time and may be defined as cursor control keys at another time. However, such input devices do not permit the array itself to be reconfigured to provide an entirely different function, such as a mouse. Also, the input device could not be reconfigured to change the position or orientation of the array on the device.

Other desirable functions such as inexpensive, simple, and reliable security for an unsecured office or laboratory are unavailable. One possible example of a need for such security would be the ability to monitor the presence of objects on a laboratory surface. For example, it may be important to know whether chemicals or experiments in a laboratory have been touched, moved, or otherwise tampered. Similarly, it may be important to know whether objects in an office were touched or moved.

Security measures limiting access to secured areas or to computer systems also may be improved. Security measures such as combination locks and computer passwords may not provide the level of security desired. For example, it may be desirable to replace or augment combination locks and passwords with other security measures that are not easily circumvented such as by discovery by unauthorized personnel.

Summary of the Invention

Tactile sensing arrays may be configured and monitored to create "intelligent" work surfaces having the versatility and adaptability to be configured to receive information and to implement various input and telecommunications devices. Intelligent work surfaces can be reconfigured at will to implement different devices. Moreover, these surfaces can be reconfigured to change the location and orientation of the devices.

Intelligent work surfaces can be implemented as intelligent desktops, floors, walls, or other such surfaces. In a preferred embodiment, the intelligent surface is a desktop surface capable of implementing input and telecommunications devices such as a keyboard, numeric keypad, mouse, and the like. An intelligent desktop also can monitor the presence and location of items on the desktop. Inputs to the intelligent surface produce outputs from the tactile array which are interpreted as outputs from the implemented device. Devices implemented on the intelligent desktop surface preferably are implemented as templates, and therefore do not occupy and clutter valuable workspace. Implemented devices may be arranged and rearranged on the work surface to suit different users and changing workspace requirements. There are no computer cables to tangle or to restrict device placement.

In another embodiment, intelligent work surfaces are implemented as floor surfaces. Intelligent floors are capable of detecting the presence and tracking the location of objects and people on the surface. In still other embodiments, the surface is configured to produce a two-dimensional image of the forces applied to the surface (a "force-image") for identifying a person. For example, the force-image of a person's handprint can be sampled, analyzed, and compared against data representing the force-images of known individuals to identify the person creating the handprint.

Brief Description of the Drawings

The above and other advantages of the invention will be apparent upon consideration of the following detailed description taken in conjunction with the accompanying drawings in which like reference numbers refer to like parts throughout and in which:

FIGs. 1 and 2 are perspective views of a desktop surface covered with sensing arrays configured as an intelligent work surface in accordance with the principles of the invention;
 FIG. 3 is a simplified, partial exploded view of the intelligent work surface of FIG. 2;
 FIG. 4 is a block diagram of circuitry for implementing intelligent work surfaces according

to the principles of the present invention;

FIG. 5 is a flowchart for implementing a keyboard-type interface on an intelligent desktop surface;

FIG. 6 is a flowchart for implementing a mouse-type interface on an intelligent desktop surface;

FIG. 7 is a flowchart for implementing an image-type interface on an intelligent desktop surface;

FIG. 8 is a flowchart for identifying an individual on the basis of a handprint by using an intelligent desktop surface;

FIGs. 9 and 10 are plan views of an image of a handprint generated on the intelligent work surface;

FIG. 11 is a flowchart for controlling a mobile vehicle using an intelligent floor surface;

FIG. 12 is a flowchart for using an intelligent floor surface to identify an individual using footprint analysis; and

FIG. 13 is a graph of data from successive footprint image samples.

Detailed Description

Tactile sensing arrays may be configured and monitored to create "intelligent" work surfaces to implement various input and telecommunications devices, such as a keyboard, a numeric keypad, a mouse, or the like. Inputs to the intelligent surface produce outputs from the tactile array which are interpreted by a processor as outputs from the implemented device. For example, force applied to the tactile array produces an electrical signal at the outputs of the array. The processor then interprets the electrical signal as a keystroke output from a keyboard. When a particular device is no longer needed or when a new device is desired, the intelligent work surface is reconfigured to implement different devices. For example, the intelligent work surface can be reconfigured such that outputs from the tactile array in the example above are interpreted as outputs from a mouse to a display screen instead of keystrokes from a keyboard.

Implemented devices may be arranged and rearranged on the work surface to suit different users and changing workspace requirements. For example, a keyboard is configured at a first location by a first computer user. The location of the keyboard is shifted, for example, to the left to accommodate the preferences of a subsequent computer user.

FIG. 1 shows a desk entirely covered with a tactile array surface 1 and a processor 2. FIG. 1 also shows a keyboard 3, a numeric keypad 4, an input interface 5, and a display screen 6. Tactile array 1 is coupled to processor 2 to form the intelligent work surface system. Tactile array 1 outputs to the processor information about inputs

applied to the array. Groups of sensing elements (see FIG. 3) which make up the tactile array are configured as an input interface to implement various devices. For example, groups of sensing elements of tactile array 1 are configured as a keyboard interface 3. As used herein, the term "region" refers collectively to a group of sensing elements on array 1. In a typical application, force applied to predetermined sensing elements within the region of the tactile array configured as keyboard 3 will generate signals indicative of keystrokes. Similarly, other groups of sensing elements of tactile array 1 are defined, for example, to be numeric keypad 4 or to be input interface 5 for display screen 6.

Input interfaces or devices implemented on tactile array 1 can be reconfigured at will to change the location, size, or orientation of the devices. Also, new devices may be configured on the tactile array and existing devices may be removed. For example, FIG. 2 shows keyboard 3, keypad 4, and display input interface 5 reconfigured to have a new location on tactile array 1. Reference numbers 3', 4', and 5' indicate the prior location (shown in phantom) corresponding to the position shown in FIG. 1, of each device. The groups of sensing elements identified by 3', 4', and 5' have been reset to their original state (which typically may be undefined). The portion of the tactile array having sensing elements 3' which overlaps with keyboard 3 is redefined to perform the appropriate functions for keyboard 3.

FIG. 3 shows the sensing elements 7 which form tactile array 1. Sensing elements 7 are formed by the intersection of rows and columns of electrical conductors. Inputs (i.e., forces) applied to tactile array 1 cause the row and column conductors to contact. Contact of the conductors permits electrical current to flow, thus generating an output from the sensing element located at the intersection of the row and column conductors.

As shown in FIG. 3, region of tactile array 1 defined by processor 2 as numeric keypad 4 overlays selected ones of sensing elements 7. Processor 2 performs a "template" function by defining a group of sensing elements as an input device, in this case, the numeric keypad. Processor 2 also performs a "mapping" function to interpret the outputs of the sensing elements as outputs of the defined input device. For example, keys "1" and "4" of keypad 4 overlay the groups of sensing elements which are designated collectively by reference numbers 8 and 9, respectively. Processor 2 will interpret outputs from sensing elements 8 and 9 (resulting from force applied to those elements) as outputs of keypad 4.

Numeric keypad 4 may be repositioned by changing in processor 2 the definition of the region

on tactile array 1 which corresponds to the keypad. Redefining the region is accomplished in this example by specifying different ones of sensing elements 7 to function as the keys of keypad 4. For example, keypad 4 may be redefined by processor 2 such that key "1" of the keypad overlays the group of sensing elements designated by reference number 9 (i.e., instead of the sensing elements designated by reference number 8, as shown in FIG. 3). Keypad 4 is shifted downward so that key "1" occupies the location formerly occupied by key "4," key "2" occupies the location formerly occupied by key "5," etc. As redefined, processor 2 will interpret the outputs of sensing elements 9 as an output "1" from keypad 4, instead of the output "4" as originally define

FIG. 4 shows several tactile sensor arrays 12 connected in parallel, which may be used to implement an intelligent work surface (i.e., tactile array 1 of FIG. 1) or a plurality of such work surfaces. Addressing circuitry 14 is associated with each sensor array 12, for addressing the individual sensing elements (also referred to as "pixels" or "taxels") which form each array 12. Circuit 14 is shown as an application specific integrated circuit or "ASIC." Analog to digital (A/D) conversion circuitry 16 (having sample and hold capability) receives data outputs passing through addressing circuitry 14 from array 12. After a predetermined settling period (determined by the array characteristics), data pass from the A/D converter to a data bus 20 via an addressable tri-state buffer 18. Data bus 20 couples to an appropriate processor such as processor 2 in FIGs. 1 and 2 for processing data in accordance with the principles of the invention. Implementation details of the illustrative sensor array circuitry of FIG. 4 will be discussed further following a discussion of illustrative techniques for implementing intelligent work surfaces.

FIGs. 5-7 flowchart illustrative methods for implementing various desktop applications in processor 2. FIGs. 11 and 12 flowchart illustrative methods for implementing various intelligent floor applications in a processor such as processor 2.

Before describing the various applications in detail, two image processing techniques common to most of the applications will be described. These techniques are "connected region analysis" (or "clustering") and determining the centroid of an image. Clustering involves grouping sampled pixels of array 12 into connected regions. Clustering techniques are well-known in the art of image processing, and are described in T. Pavlides, Algorithms for Graphics and Image Processing, Comp. Science Press, 1982. The centroid of each clustered region may be determined for an $n \times m$ sensor array using the formulas:

$$\bar{x} = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m i * V_{ij}$$

$$\bar{y} = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m j * V_{ij}$$

where n and m are the numbers of rows and columns in the sensor array, respectively, i and j are the row and column displacement, respectively, and V is the gray scale value of the pixel at coordinates (i,j) .

In a desktop application, data typically will be input to a processor through one of three types of interfaces: a "keyboard" interface, a "mouse" interface, or an "image" interface. Portions of the desktop surface can be defined, through an initialization procedure, to be input devices of one of these types of interfaces. The processor defines input devices by the template and mapping functions described above. That is, the processor first identifies selected sensing elements as an input device. The processor then interprets (maps) the outputs of the selected sensing elements to be outputs of the input device. The processor may define several input devices simultaneously, on different portions of the desktop.

Referring again to the figures, the flowchart of FIG. 5 shows an illustrative method for implementing a keyboard interface. The keyboard interface may be used to provide keyboards (such as keyboard 3 of FIG. 1), numeric keypads, and other similar devices. (These devices may be constructed, for example, by overlaying a sheet of compliant material with a patterned template.) As will be described more fully below, the process illustrated by FIG. 5 also is used to define other, non-keyboard devices.

As shown in box 30, a processor (such as processor 2 of FIG. 1) samples sensor array 12. This may be done, for example, by cycling through the array to address each sensing element. Sampled points then are "clustered" into connected regions, and the processor computes the centroid of each region. Once the data is collected from the array, the processor performs one of three general operations. The processor either defines a device, performs a processing operation for a device already defined, or performs an "optional" or special processing operation upon the occurrence of some predetermined event (e.g., contact to tactile array 1 outside of any defined region).

Boxes 32, 34, 36, and 38 show how a module (i.e., a defined area on the sensor array or a device such as a keyboard or mouse) may be added to a

list of "active" devices. The array is sampled until at least one cluster is detected. Upon detecting a cluster, the processor determines whether there are three clusters. If three clusters are detected, the processor determines whether the pattern formed by the centroids of the three clusters correspond to the "footprint" of a known module. If so, the module and its position and orientation are added to a list of active devices. If the centroids do not correspond to the footprint of a known module, the inputs may be ignored, and the array is sampled again.

Once a device is active, the processor monitors arrays to determine, for example, whether a key is struck in the active region. If the response to decision box 34 is "no" (i.e., the number of clusters does not equal three), the processor determines whether there is only one cluster (box 40) and to which active device this cluster belongs (box 44). Because each device is divided into non-overlapping regions, the centroid of a single cluster will reside in one region identifying, for example, the "key" of the keyboard that was pushed. Device-specific software running on processor 2 then interprets the outputs of the tactile array as outputs of the defined input device and executes functions on the basis of the detected keystrokes (box 46).

If the decision at box 40 is "no," (i.e., there is more than one cluster) the processor preferably performs additional processing to act on multiple contacts to the sensor array (box 48). This provides a means for allowing module-independent functions, such as system on/off, last entry erasure, session termination or restart, or region definition for other module types (e.g., a mouse). Similarly, if the answer to decision box to 42 is "yes," the processor preferably performs additional processing (e.g., erasing previous key strokes of a calculation by striking outside an active calculator area) for contacts outside the active region of any known module (box 50).

An illustrative flowchart for implementing on tactile array 1 a second interface type, a mouse device, is shown in FIG. 6. The mouse active region (i.e., the sensing elements which define the bounds of the device) must first be defined (box 50) on tactile array 1. This is done using the steps illustrated by the flowchart of FIG. 5. The mouse is then implemented by performing three distinct operations. First, an origin must be established in the mouse region. The origin establishes a position of a cursor on the screen of a display monitor. Second, click events must be detected and indicated. A click event is an input to the mouse device which indicates that a predetermined processing operation should occur. Third, the cursor on the screen of the display monitor must be repositioned in

response to repositioning of the origin in the mouse region.

Having defined the mouse active region, the tactile array is sampled until a cluster is detected in the defined mouse region (box 52). Once a cluster is detected, the processor determines whether an origin has been established (box 56). If an origin has not been established, processing continues as shown in box 58. The processor samples the mouse region of the sensor array several times. The average centroid is calculated by adding the centroid x and y values to a running total, and dividing that running total by the number of samples taken. An origin is then marks A sample counter may be used to ensure that a predetermined number of samples are taken (box 60). The number of samples taken should exceed the number required to reliably locate the origin of original contact. If the original cluster is not in the active mouse region, the running total counters are reset (box 54), the origin is not established, and the next sample is taken.

Once an origin is established as described above, the processor continues to sample the array. The processor determines for subsequent samples whether the sum of the pixel values in the cluster of each sample exceeds the "click threshold" (see box 64). The click threshold is a predetermined value over which it is assumed that a mouse "click" event has occurred. Exceeding the click threshold has the same effect as depressing the buttons of a conventional mouse. Upon occurrence of a click event, the processor executes instructions as required by a particular application on occurrence of a click event (box 66). If the sum of pixel values in a cluster does not exceed the click threshold, the processor determines whether the centroid of the cluster has been moved a predetermined distance (the "origin distance") from the centroid of the origin (box 68). If it has not, the array is sampled again. If the centroid has moved a sufficient distance, a cursor displayed on the screen of an associated display monitor (not shown) is moved to a new position prior to taking the next sample (box 70). The cursor on the display monitor is moved in the direction from the origin to the current centroid. The cursor is moved a distance equal to the product of a "velocity constant" and the distance between the origin and the current centroid. The velocity constant depends upon the sensor resolution, the resolution of the display monitor, and the sampling rate, and preferably may be changed by a user to suit the user's preference as to how fast the cursor should move. Techniques for moving the cursor in response to the new centroid are similar to techniques for moving a display cursor in response to movement of a conventional mouse. These techniques are well-

known in the art.

The image interface is described with reference to FIG. 7. When processing images, a sensor array is sampled (box 72) and the gray scale pixel values (or other multi-scale values) from the array are stored as a image (i.e., each pixel having a row and column address)(box 74). The image gray scale data is then processed as required for a specific application (box 76). One such application is described with reference to FIGs. 8-10.

FIG. 8 shows an illustrative method for identifying an individual on the basis of a handprint. This application can be used, for example, to grant or deny to an individual access to secure areas or to computer terminals. The first step in the process, shown in box 78, is to obtain successive array samples, and to sum those samples to produce a full image of a handprint. The processor typically samples the sensing array at a rate such as 50 Hz, making it desirable to sum the samples to provide a reliable image.

The image then may be thresholded (box 80) by finding the value of all non-zero pixels, and subtracting a predetermined value (e.g., 0.3 times the average pixel value) from each non-zero pixel. Following the subtraction, thresholding continues by setting any pixel having a value less than zero to zero. This process eliminates noise, and permits a person to hold a hand against the sensing array for any period of time to facilitate clustering by the processor.

The next step (box 82) is to convert the gray scale handprint image to a binary image. All non-zero pixels are given a value of one. This gives equal weight to each pixel, and provides a better unit of measure for subsequent operations than that available using gray scale centroid. FIG. 9 shows a handprint image 106 resulting from this process. The remaining steps of the method are described with reference to FIGs. 9 and 10.

The next step (box 84 of FIG. 8) is to locate the centroid 108 and the principle axes of binary image 106. Centroid 108 is determined using the centroid formulas described with respect to clustering above, where the value of all non-zero pixels equals one. The X-axis 110 of a quadrant frame with an origin at centroid 108 is parallel to the principle axes corresponding to the lesser principle moment, toward the heel of the palm. The Y-axis 112 is perpendicular to X-axis 110, completing a right-handed coordinate frame. This identifies the orientation of the hand.

The binary image is clustered into connected regions (box 86), and the centroid of each connected region is determined. The cluster 114 corresponding to the tip of the thumb then is located (box 88). Cluster 114 is the cluster having the centroid which is furthest from X-axis 110, and

whose centroid has an x coordinate of between -1.2 and 3.5 inches.

Next, handedness is determined. If the y coordinate of the thumb centroid is positive, the handprint image is of a left hand. If the y coordinate is negative, it is the right hand

The clusters corresponding to the fingertips then are identified and ordered (boxes 90 and 92). To do this, the processor computes the distance between centroid 108 and the centroid of each cluster that has a centroid with a negative x coordinate (excluding the thumb). The processor adds to this value the square of the x coordinate of the centroid. The four centroids with the greatest values are the fingertips. The fingertips are then ordered according to their y coordinate values, including the thumb. If the handprint is of a left hand, the thumb will have the greatest y coordinate, the index finger the next highest, followed by the middle ring and little fingers. If it is a print of a right hand, the ordering is reversed

As shown in box 94, the processor then locates the furthest pixel from centroid 108 in each of fingertip clusters 114, 116, 118, 120, and 122.

Next, the center-bottom point 124 of the palm of the hand is located (box 96). To do this, a new coordinate frame, shown in FIG. 10, is defined. The new coordinate frame is based at the same origin as the original coordinate frame. The new X-axis 136 is defined to be parallel to a line connecting the furthest pixel away from the centroid 126 of the middle finger cluster 120 and the midpoint 128 of a line segment drawn between the centroid 130 of thumb cluster 114 to the centroid 132 of little finger cluster 116. The new Y-axis 134 completes a right-handed coordinate frame. Using this coordinate frame, the x coordinate of bottom-center point 124 is the x coordinate of the non-zero pixel with the greatest x value in the new coordinate frame. The y coordinate is determined by finding the two non-zero pixels, with a x coordinate greater than zero, that have the greatest and smallest y coordinates, respectively (excluding pixels in thumb cluster 114). The average of these two y coordinates is the y coordinate of center-bottom point 124.

The processor then determines the palm width of the hand (box 98). The palm width is defined to be the absolute distance between the two extreme y values, excluding clusters identified as fingers. The processor also determines the distances between the center-bottom of the hand 124 and the furthest pixel of each fingertip (box 100).

A six-tuple is used to characterize an individual's handprint (see box 102). The six values that form the six-tuple are the distances from bottom-center point 124 to the furthest pixel in each fingertip, and the palm width. This six-tuple can be compared to a library of six-tuples which define a

library of handprints (box 104). Thus, a handprint from an unknown individual can be identified by calculating the six-tuple defined as $\langle d_1, \dots, d_6 \rangle$, and comparing it to the six-tuples previously stored for known individuals (defined as $\langle C_{x1}, \dots, C_{x6} \rangle$). A comparison is made by computing the euclidean distance between tuples for each known individual according to the formula:

$$D_x = \left(\sum_{i=1}^6 (C_{x,i} - d_i)^2 \right)^{1/2}$$

The unknown handprint is identified as belonging to the known individual with the lowest D_x value.

The flowcharts of FIGs. 11 and 12 illustrate methods for implementing in processor 2 various intelligent floor applications.

FIG. 11 flowcharts a method for recognizing and controlling a motorized vehicle operating on a surface (such as a floor) that is covered with sensor arrays. The vehicle is identified and its position is determined. The vehicle is then driven to a desired location under feedback control. Collisions between the vehicle and other objects are avoided using a priority determining procedure.

The first step in controlling a motorized vehicle is to sample the floor sensor array, as shown in box 140. The process then identifies the vehicle by its known footprint (box 142). To do this, the processor performs cluster analysis as described above, determines the centroids of the detected clusters and any distances between the centroids, and compares this information to the known footprint of the vehicle. The processor then determines the location of the vehicle on the floor (box 144), and compares the location of the vehicle to the desired location. The vehicle's previous location, once available, can be used to limit the search for the vehicle's current location. The motor and steering units of the vehicle are then actuated (as required for the type of vehicle being operated) to drive the vehicle toward the desired location. The processor repeatedly samples the sensing arrays on the floor and adjusts the motor and steering units (box 146) until the vehicle arrives at the desired location.

The processor preferably provides a means for avoiding collisions with obstacles, such as other vehicles or people (see box 141). In one embodiment, if the vehicle comes within a predetermined distance of another vehicle (or an object), the relative priorities of the two vehicles are determined (box 143). The vehicle having the higher priority is driven to the desired location (box 145). The lower-priority vehicle remains stopped until the higher-

priority vehicle has moved out of range (box 147), and then continues toward the desired location. Vehicles typically are given higher priority than stationary objects, yet lower priority than people. The direction and speed of the vehicle can be modified as required to avoid obstacles and depending on traffic patterns on the floor.

FIG. 12 flowcharts a method for identifying an individual on the basis of footprint analysis. The first element of the footprint analysis is to sample the array as shown in box 148. The footprint must be sampled at a constant rate so that a template of the footprint, comprising a series of footprint images, may be constructed. Once constructed, the template describing the "unknown" (i.e., as yet unidentified) footprint is compared to a library of templates of "known" footsteps, so that the unknown footprint may be identified.

The processor computes the "origin" of the footprint (box 150) preferably by defining the first frame (the first footprint image of the footprint) as the sensor array sample that detects the heel of the footprint contacting the floor. The centroids of this first footprint image and a second footprint image are computed and averaged. This average centroid defines the origin.

For each succeeding footprint image, until the toe lifts off the floor at the end of the footprint, the processor samples the sensing array, and computes and stores the centroid of the footprint image (boxes 152, 154, and 156). The processor also computes and stores the distance from the centroid of each footprint image to the origin (box 158), and the sum of all pixel values (box 160). The sum of all pixel values is proportional to the weight applied to the array. Once all footprint samples have been taken (i.e., the foot no longer contacts the floor), the processor reinterpolates the data (box 162).

If the processor took N samples (i.e., N footprint images) during a footprint, there will be $N - 2$ computed distances and pixel sums (i.e., weights). As shown in FIG. 13, this method focuses on the relationship between distance versus weight. The graph of FIG. 13 is derived from the weight versus time and distance versus time information collected through the series of array samples taken from the time the heel of the foot contacts the sensor array to the time the toe is lifted from the sensor. To obtain data which is evenly spaced on the abscissa, the data is reinterpolated, preferably at intervals of $15/128$ distance units. This results in 128 data points between 0 to 15 distance units (i.e., inches). (The number 15 is chosen here because the largest expected footprint will not exceed 15 inches in length.) These 128 data points (some of which may be zero) define a 128-tuple $\langle d_1, \dots, d_{128} \rangle$, which characterizes the unknown footprint (see box 164 of FIG. 12). The 128-tuple of the unknown

footstep may be compared with the 128-tuple of known footsteps $\langle\langle C_{x1}, \dots, C_{x128} \rangle\rangle$, for individual "x" from a library of footsteps (box 166 of FIG. 12). The footsteps may be compared using a Euclidean distance measure:

$$D_x = \left(\sum_{i=1}^{128} (d_i - C_{x,i})^2 \right)^{1/2}$$

The unknown footstep (and the unknown individual to whom the footstep belongs) is identified as the footstep (and individual) having the smallest D_x value.

Referring again to FIG. 4, details of illustrative circuitry for implementing the intelligent work surfaces will now be described. Sensor arrays 12 preferably cover a work surface without significantly altering height or profile of the surface. Each of sensor arrays 12 typically may be a piezoresistive tactile sensor array, such as a polymer ink array. Sensor arrays 12 have separately addressable sensing elements, which typically comprise intersections of rows and columns of conductive elements. Thus, array 12 may be "sampled" by sequentially addressing the columns and rows. Sensor arrays suitable for use as array 12 are commercially available from Interlink Electronics, Carpinteria, California 93013. Of course, one skilled in the art will appreciate that many other types of sensors, such as variable capacitance sensors, may be used to implement sensor array 12, without departing from the scope or spirit of the invention.

Addressing circuitry 14 typically may include circuitry for isolating each sensing element in array 12, to prevent undesirable current flow through parallel paths in the array. Circuitry for implementing addressing circuitry 14 is described in commonly owned, co-pending, U.S. patent application Serial No. 07/775,881, filed on October 11, 1991, which is hereby incorporated by reference herein in its entirety. A/D converter 16 may be implemented using any commercially-available A/D converter circuit.

Addressing circuitry 14, A/D converter 16, and buffer 18 are controlled via control lines 22. Control signals are broadcast over lines 22 to all sensor arrays 12 simultaneously, and data is sequenced onto common data bus 20 via buffers 18. A strobe signal times the conversion of the analog output from sensor array 12 (through addressing circuitry 14) by A/D converter 16.

Any number of sensor arrays 12 (up to the limits of the sensor select code) can be accessed at the same rate as a single array, provided the time required to select an array and output the data

from the array to bus 20 is less than the settling period. If more than 2^8 sensor arrays 12 are required, the data bus can be enlarged to accommodate additional sensors.

The primary parameters to be determined in an intelligent work surface application are sensing element density and array sampling frequency. These parameters are controlled largely by the application. For example, intelligent desktops generally require greater sensing element density than intelligent floors.

Assuming a 32 microsecond settling period, an 800 nsec strobe (with data ready at the output 800 nsec after the strobe, as is typical of commercially available A/D converters) and a 1 microsecond addressing period, a sensing element of the array can be addressed and sampled, on average, every 33.8 microseconds. Thus, an array of 16 x 16 sensors (each array having 256 sensing elements), can generate data at a rate of 7.54 megapixels/second. This data rate is similar to that of a standard 512 x 480 pixel video image, which generates 7.37 megapixels/second when updated at 30 Hz.

To implement typical modular input devices, such as a keyboard or mouse, sensing element densities of 0.125 to 0.25 in² typically are adequate. At this resolution, an intelligent desktop generates data at rates from 1.2 to 4.9 megapixels/sec. This is within an order of magnitude of video images at 30 Hz.

An area of 12' by 12' is typical for an office or small common equipment area in an office building. In an area of this size, with sensing element densities ranging from 1.5 in² down to 0.25 in², data can be generated at a rate of 0.27 Mpixels/sec to 9.78 Mpixels/sec. In the midrange, using a sensor density of 3/4 in², an intelligent floor generates data at 1.10 Mpixels/sec, approximately fifteen percent of that generated by a single video camera. However, these data rates can be altered substantially by increasing the sampling rate to 115 Hz, or by decreasing the array resolution as described, for example, in U.S. patent application Serial No. 07/775,881, discussed above.

It will be understood that the foregoing is merely illustrative of the principles of the invention, and that various modifications can be made by those skilled in the art without departing from the scope and spirit of the invention. For example, the sensor arrays of the present invention could be applied to wall surfaces, and used interactively and in conjunction with projected information from a computer display. An individual could respond to projected information by contacting the "intelligent wall surface," to provide input to a computer. The scope of this invention is limited only by the claims that follow.

Claims

1. A work surface system comprising:
 an array of sensing elements to form a work surface, each sensing element of the array producing a data output in response to an input to the sensing element;
 means for defining a first input device as comprising a first plurality of sensing elements;
 means for mapping each data output from the first plurality of sensing elements to a corresponding output of the first input device; and
 means for redefining the first input device as comprising a second plurality of sensing elements, wherein the means for mapping maps data outputs from the second plurality of sensing elements to a corresponding output of the first input device.
2. The apparatus of claim 1 further comprising:
 means for mapping the data outputs from at least a portion of the first plurality of sensing elements to a corresponding output of a second input device.
3. The apparatus of claim 1 further comprising:
 means for defining a second input device as comprising a third plurality of sensing elements; and
 means for mapping each data output from the third plurality of sensing elements to a corresponding output of a second input device.
4. The apparatus of claim 1 further comprising means for performing connected region analysis on the outputs from the first input device to separate the outputs into connected regions.
5. The apparatus of claim 4 further comprising means for determining the centroid of at least one connected region.
6. The apparatus of claim 1, wherein the first input device is a keyboard interface.
7. The apparatus of claim 1, wherein the first input device is a mouse interface.
8. The apparatus of claim 1, wherein the first input device comprises an image interface, and wherein the apparatus further comprises means for storing as an image a multi-level representation the inputs to the first plurality of sensing elements.
9. The apparatus of claim 1, wherein the means for defining comprises means for defining simultaneously a plurality of input devices on the array, each of the plurality of input devices comprising predetermined ones of the sensing elements.
10. A work surface system comprising:
 a array of sensing elements to form a work surface, each sensing element of the array producing a data output in response to an input to the sensing element;
 means for defining a first input device as comprising a first plurality of sensing elements;
 means for mapping each data output from the first plurality of sensing elements to a corresponding output of the first input device; and
 means for mapping the data outputs from at least a portion of first plurality of the sensing elements to a corresponding output of a second input device.
11. A method for using a sensor that has a plurality of sensing elements each of which produces a data output in response to an input, comprising the steps of:
 defining a first input device as comprising a first plurality of the sensing elements;
 mapping each data output from the first plurality of the sensing elements to a corresponding output of the first input device; and
 redefining the first input device as comprising a second plurality of the sensing elements, wherein the means for mapping maps data outputs from the second plurality of the sensing elements to a corresponding output of the first input device.
12. The method of claim 11 further comprising:
 mapping the data outputs from at least a portion of the first plurality of the sensing elements to a corresponding output of a second input device.
13. The method of claim 11 further comprising:
 defining a second input device as comprising a third plurality of the sensing elements; and
 mapping each data output from the third plurality of the sensing elements to a corresponding output of a second input device.
14. The method of claim 11, wherein the defining step comprises the steps of:
 sampling the sensing elements of the sensor;
 determining whether a predetermined pattern of force has been applied to the sensor; and

defining the location and orientation of a predetermined input device in response to the predetermined pattern of force.

15. The method of claim 11, further comprising the step of performing connected region analysis on outputs from the first input device to separate the outputs into connected regions. 5
16. The method of claim 11, further comprising the step of determining the centroid of at least one connected region. 10
17. The method of claim 11, wherein the first input device is a keyboard interface. 15
18. The method of claim 11, wherein the first input device is a mouse interface.
19. The method of claim 18, further comprising the steps of: 20
 defining an origin to be the centroid of a first force exerted within the defined region;
 determining whether the centroid of a second force is spaced apart from the origin by at least a first predetermined distance; and 25
 generating signals indicative of moving a cursor on a display screen in a predetermined direction by a second predetermined distance if the centroid of the second force is spaced apart from the origin by at least said first predetermined distance. 30
20. The method of claim 19, further comprising the steps of signaling a click event upon detecting a third force which exceeds a predetermined value. 35
21. A method for using a sensor that has a plurality of sensing elements each of which produces a data output in response to an input, comprising the steps of: 40
 defining a first input device as comprising a first plurality of the sensing elements;
 mapping each data output from the first plurality of the sensing elements to a corresponding output of the first input device; and 45
 mapping the data outputs from at least a portion of the first plurality of the sensing elements to a corresponding output of a second input device. 50
22. A method for identifying a person, comprising the steps of: 55
 sampling a tactile sensor array to record a force-image that is characteristic of the person;
 determining from the recorded force-image information characteristic of the person;

comparing the information characteristic of the person with information characteristic of known individuals; and

identifying the person as one of the known individuals on the basis of the comparing step.

23. The method of claim 22, wherein the force-image characteristic of the person comprises a force-image of a handprint.
24. The method of claim 23, wherein the sampling step comprises:
 summing pixels output from successive samples of the sensor array to produce the force-image;
 subtracting a predetermined value from each non-zero pixel;
 setting to zero all pixels having a value less than zero; and
 setting to one all non-zero pixels.
25. The method of claim 23, wherein the determining step comprises:
 defining an origin at a centroid of the force-image;
 determining a set of principle axes of the force-image;
 performing connected region analysis on the force-image to separate the force-image into a plurality of connected regions;
 identifying the connected regions corresponding to fingertips of the force-image;
 identifying a pixel of each fingertip that is spaced furthest from the origin;
 calculating the distances between a predetermined point on the force-image and the pixel of each fingertip that is spaced furthest from the origin; and
 calculating the width of the palm of the force-image.
26. The method of claim 22, wherein the force-image characteristic of the person comprises a series of sequential force-images of a footprint.
27. The method of claim 22, wherein the determining step comprises:
 defining an origin at a centroid of the first force-image of the series of sequential force-images;
 calculating for each succeeding force-image: a centroid of the force-image, the distance between the centroid of the force-image and the origin, and a sum of all pixel values of the force-image; and
 correlating for each force-image the sum and the distance between the centroid of the force-image and the origin, to characterize the

person.

- 28.** The method of claim 27, further comprising the step of reinterpolating the sum and the distance to obtain data values at predetermined intervals for characterizing the person. 5

10

15

20

25

30

35

40

45

50

55

FIG. 1

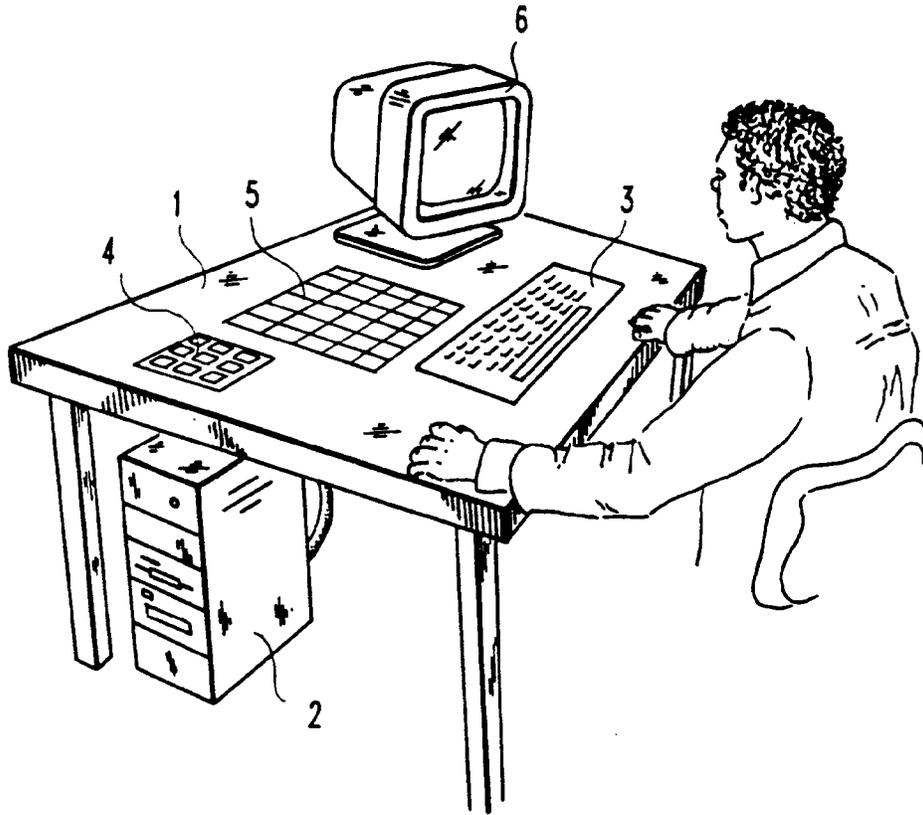


FIG. 2

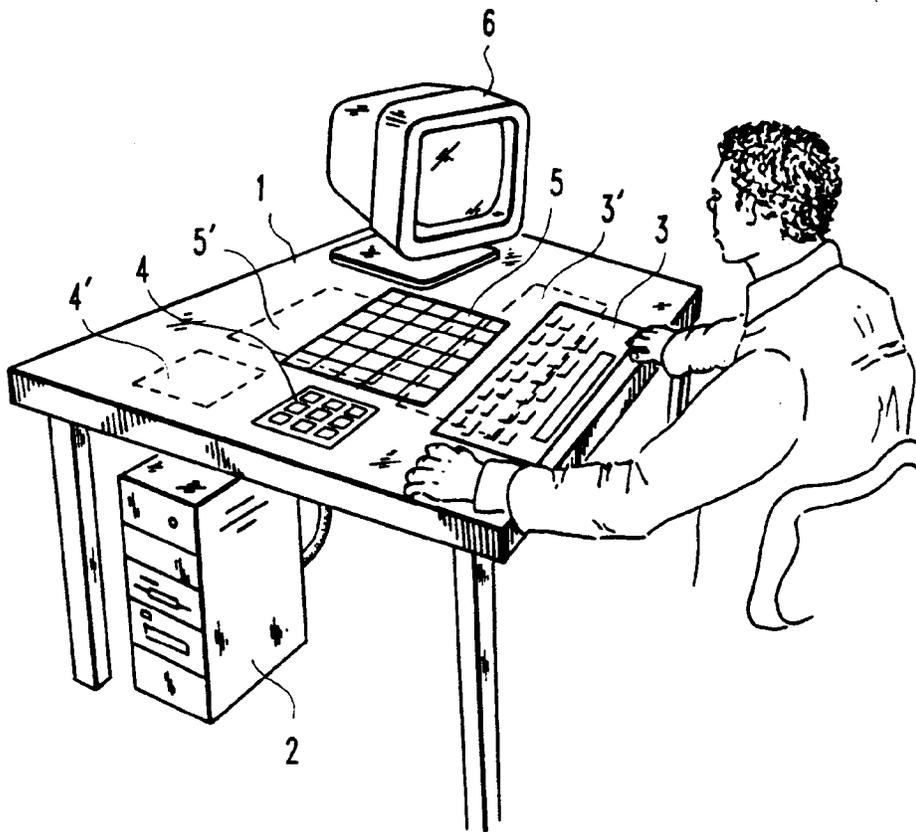


FIG. 3

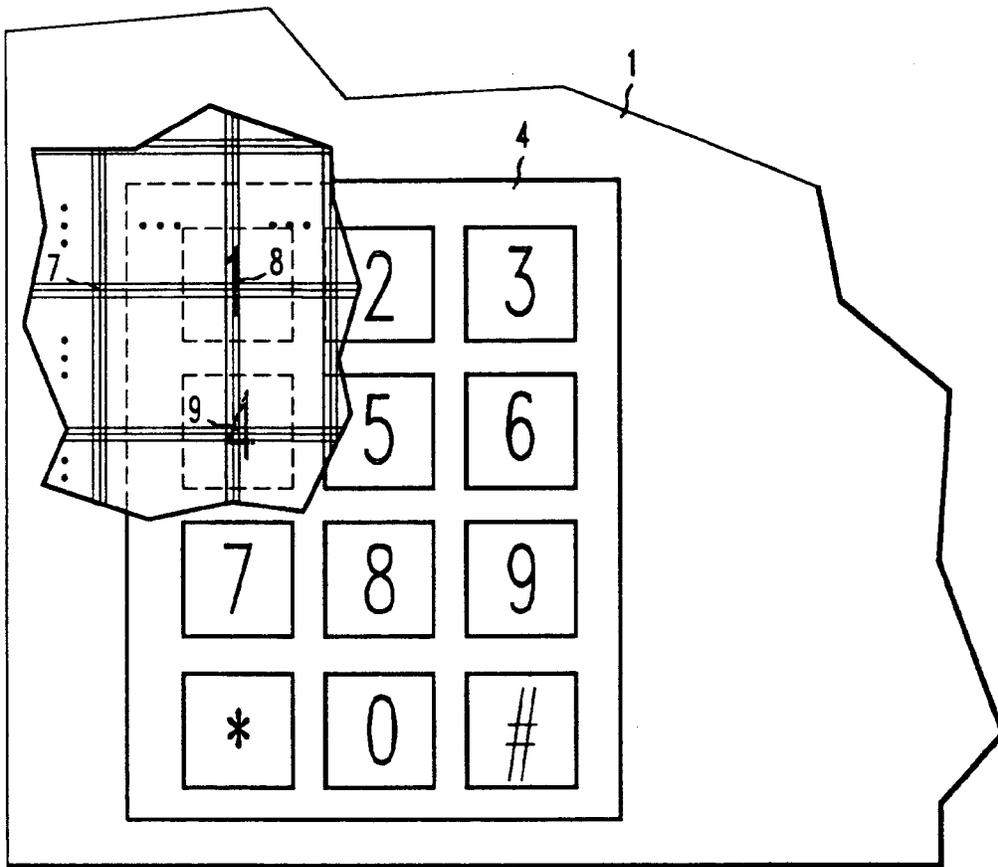


FIG. 4

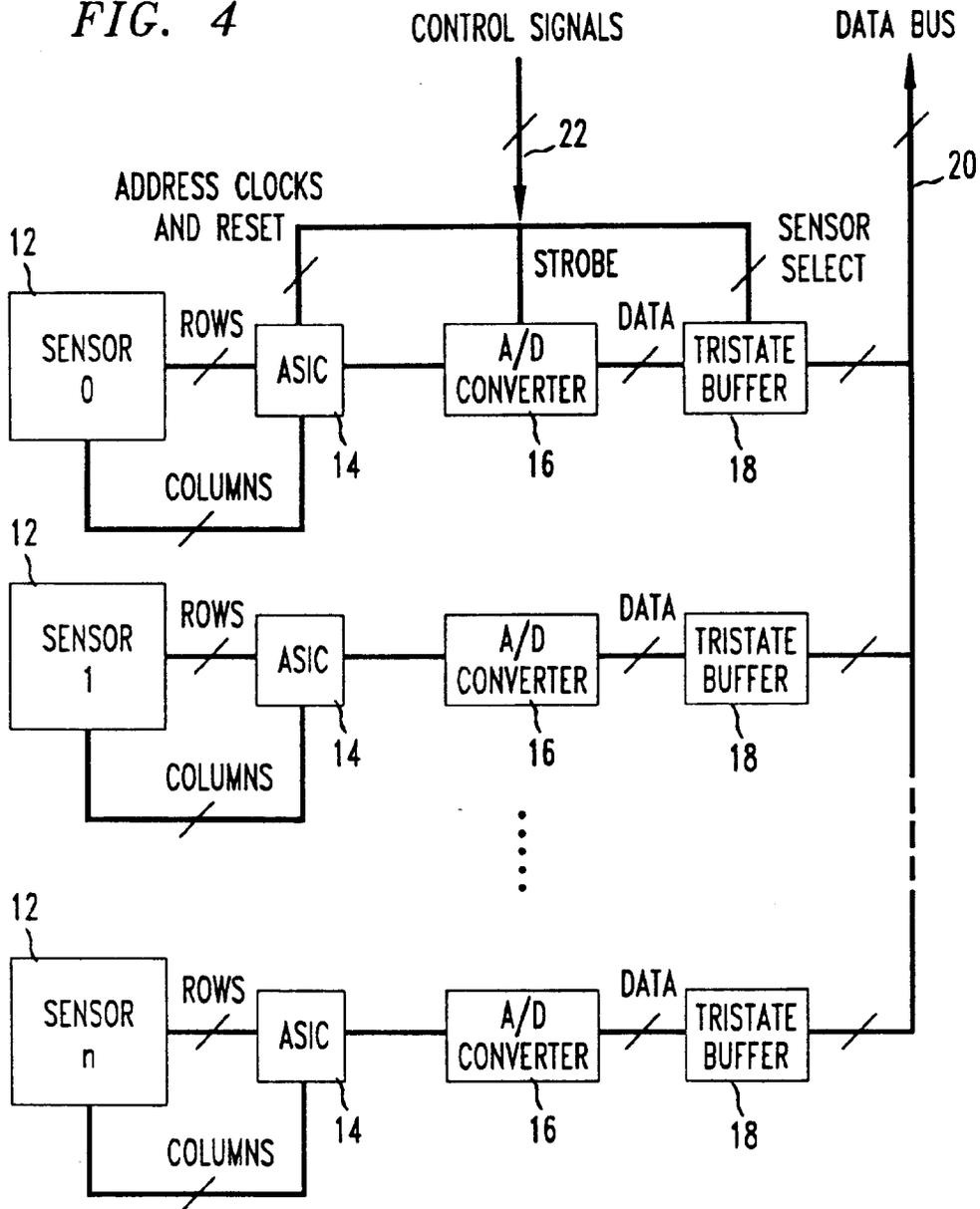


FIG. 5

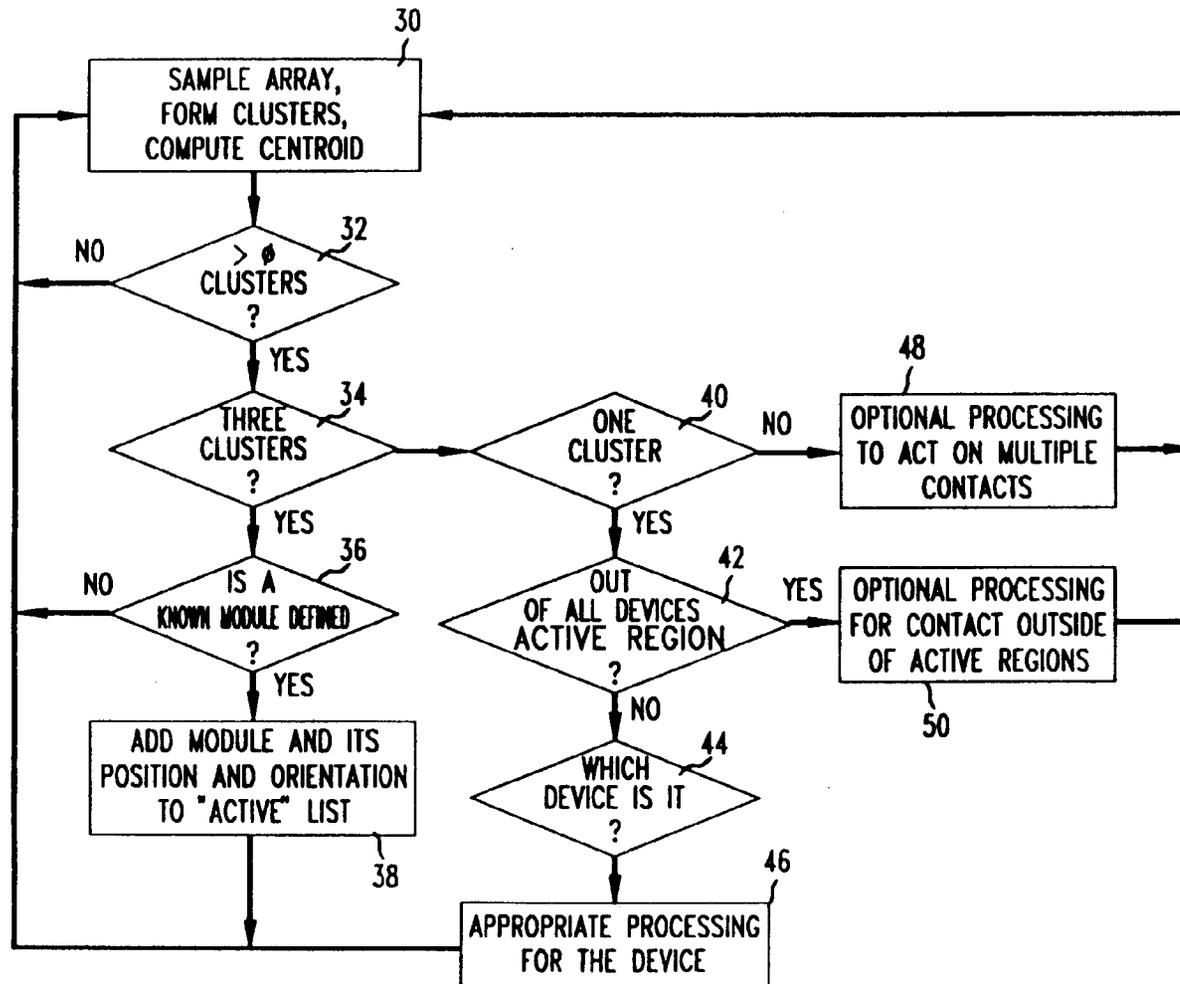


FIG. 6

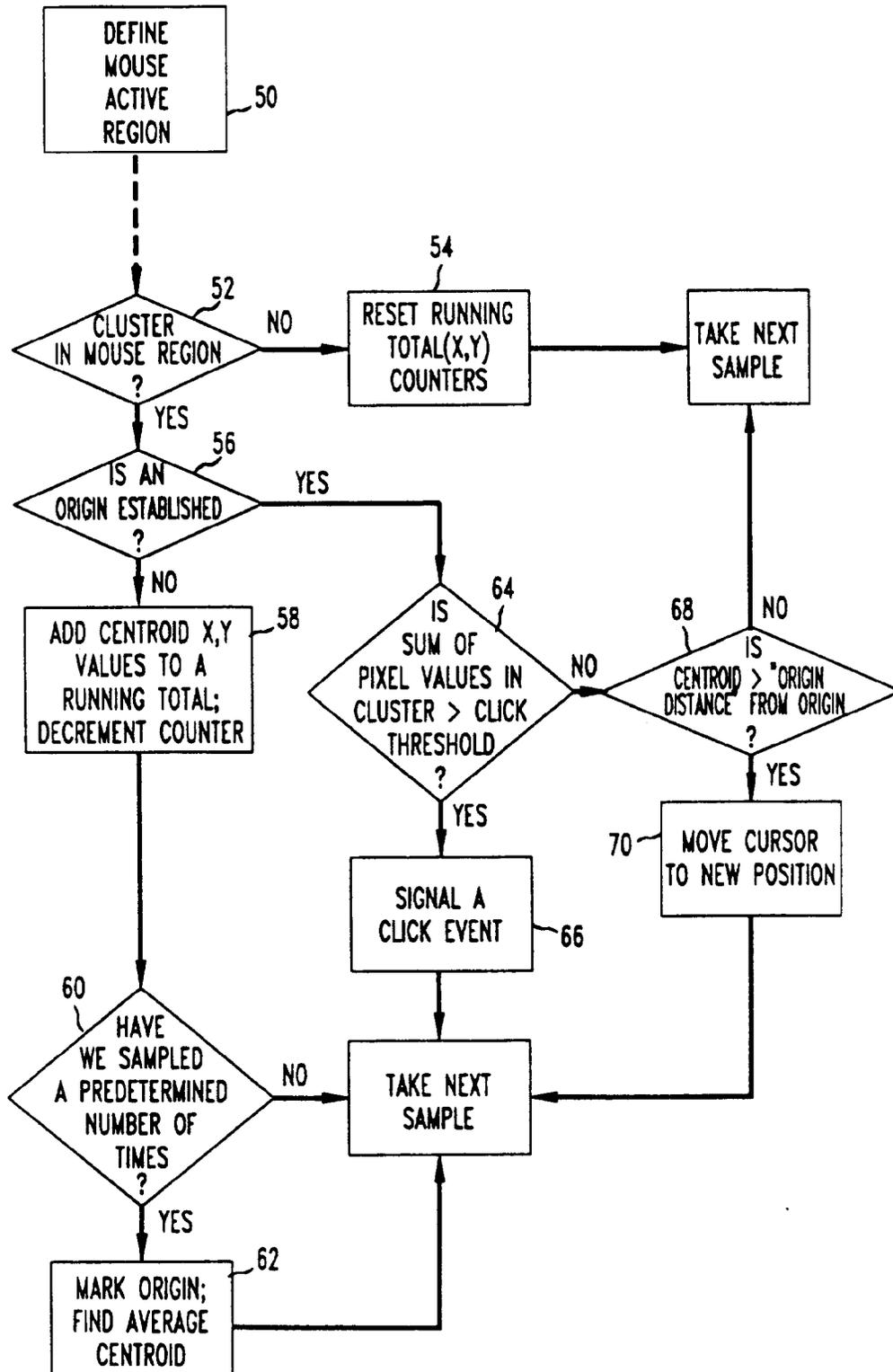


FIG. 7

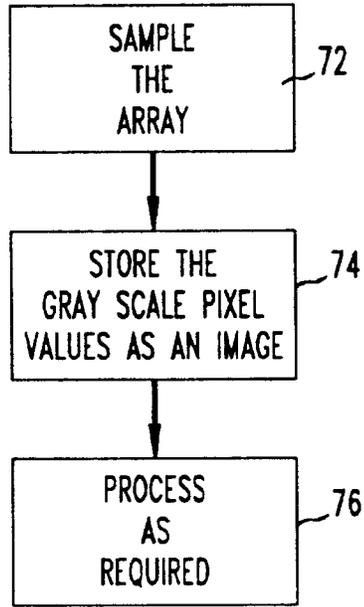


FIG. 13

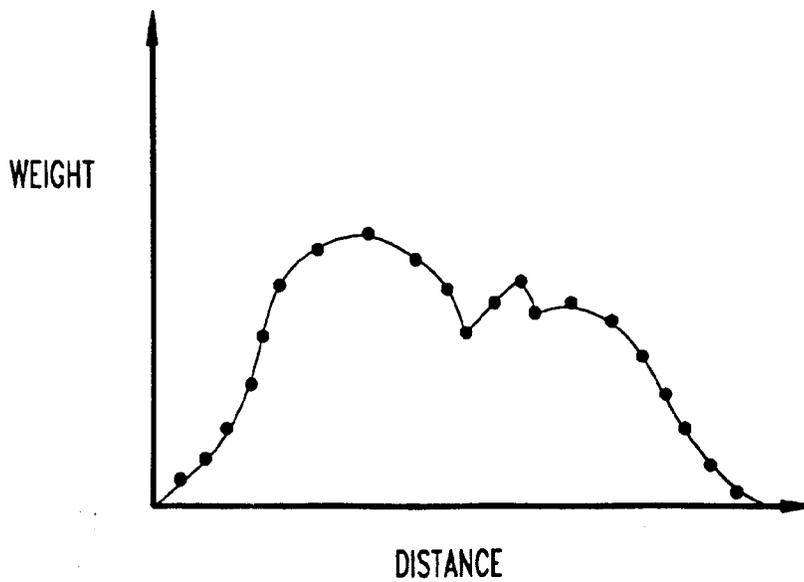


FIG. 8

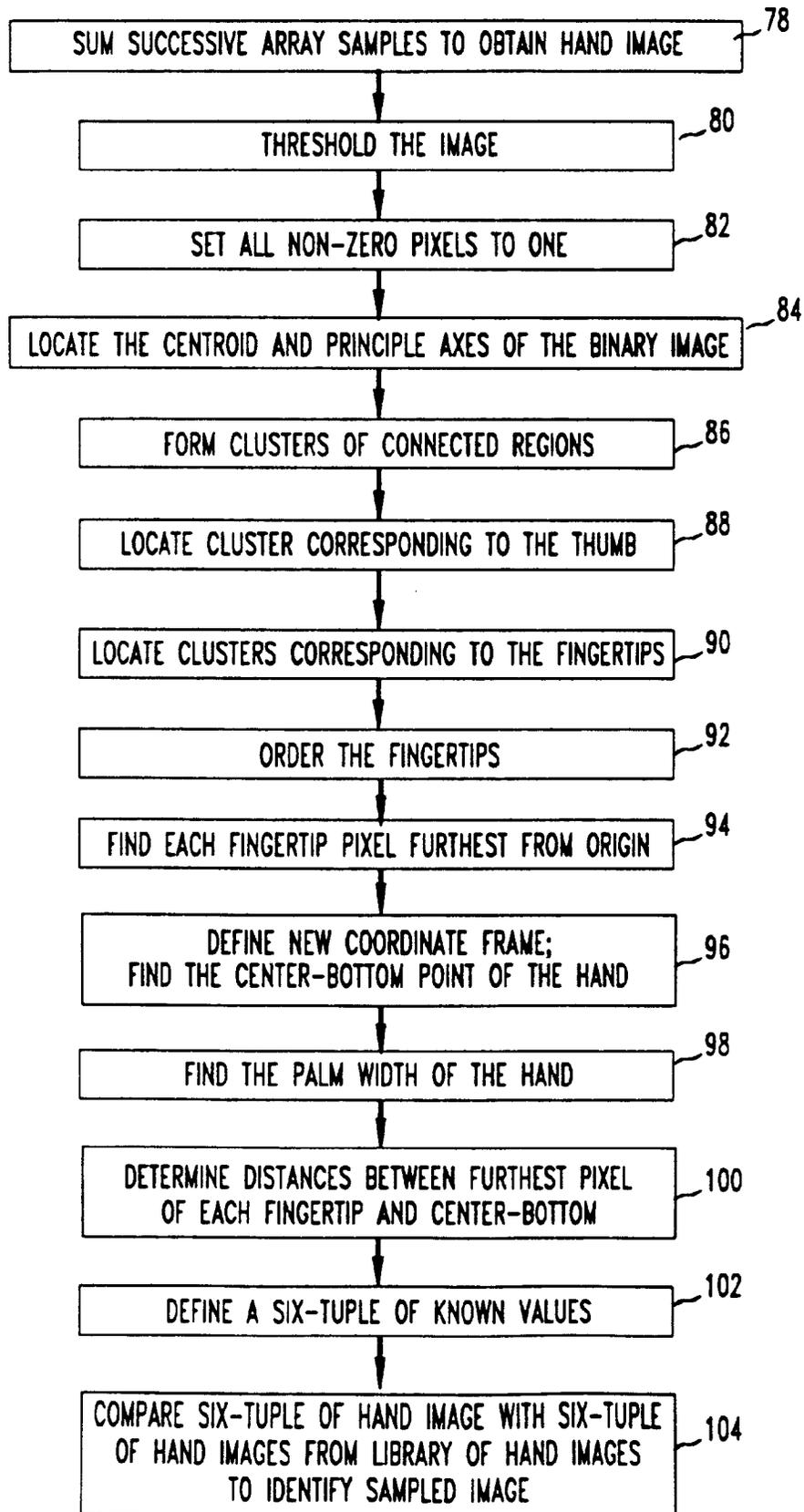


FIG. 9

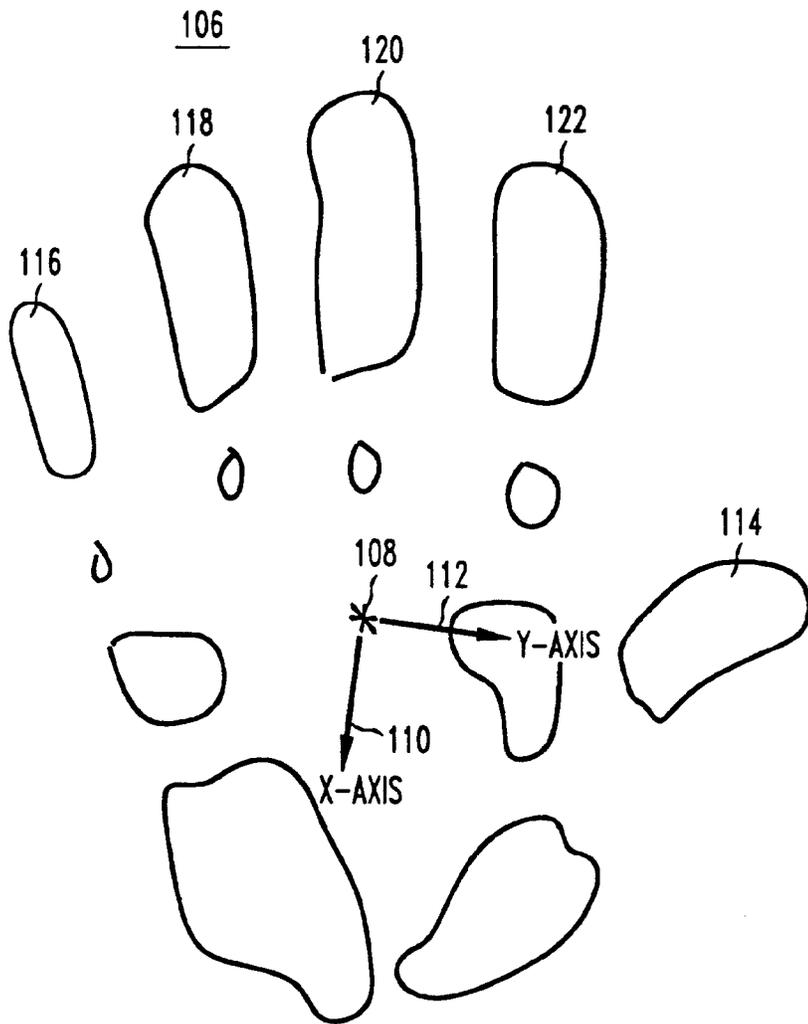


FIG. 10

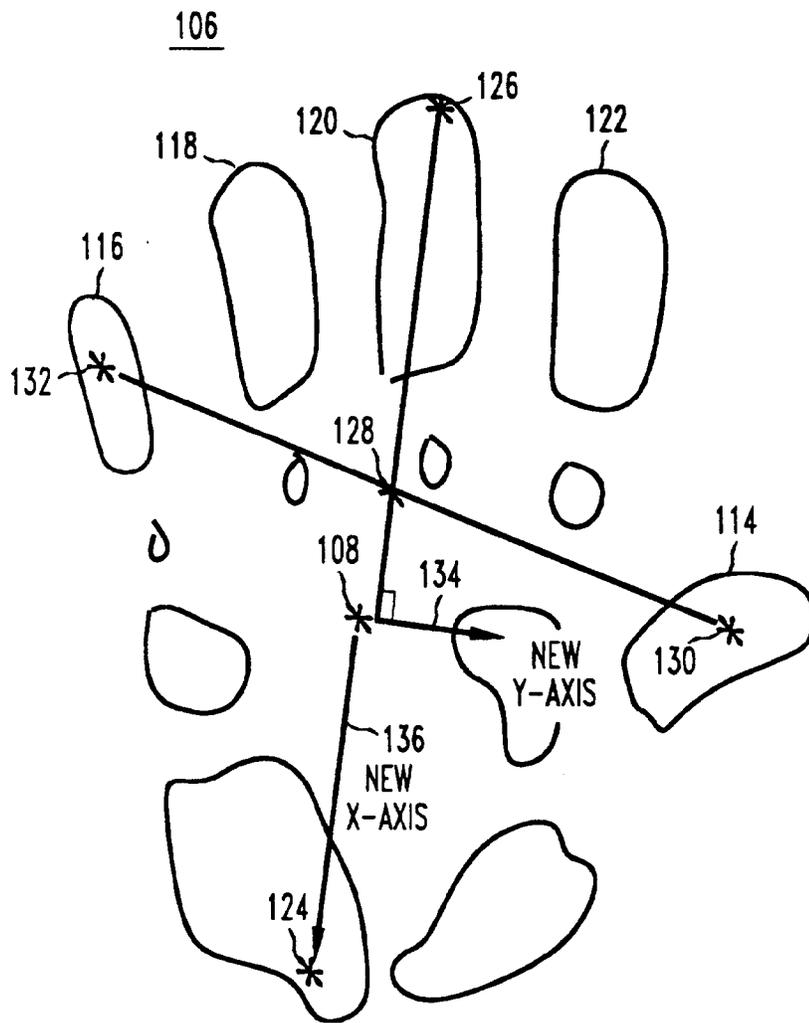


FIG. 11

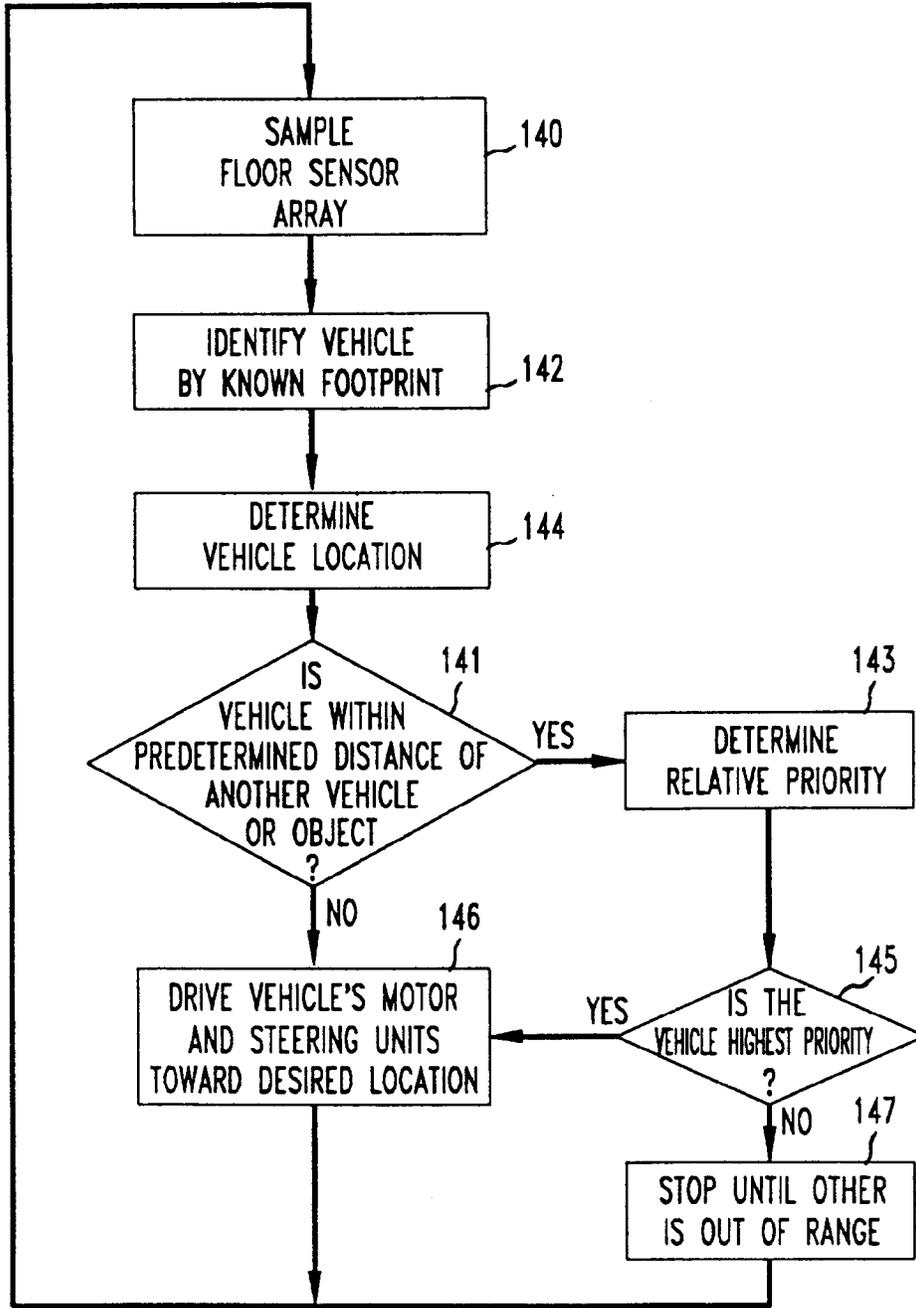


FIG. 12

