

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6570376号  
(P6570376)

(45) 発行日 令和1年9月4日(2019.9.4)

(24) 登録日 令和1年8月16日(2019.8.16)

(51) Int.Cl.

F 1

G06F 3/042 (2006.01)  
G06F 3/041 (2006.01)G06F 3/042 473  
G06F 3/041 590

請求項の数 10 (全 24 頁)

(21) 出願番号 特願2015-168293 (P2015-168293)  
 (22) 出願日 平成27年8月27日 (2015.8.27)  
 (65) 公開番号 特開2017-45332 (P2017-45332A)  
 (43) 公開日 平成29年3月2日 (2017.3.2)  
 審査請求日 平成30年8月7日 (2018.8.7)

(73) 特許権者 000001007  
 キヤノン株式会社  
 東京都大田区下丸子3丁目30番2号  
 (74) 代理人 100126240  
 弁理士 阿部 琢磨  
 (74) 代理人 100124442  
 弁理士 黒岩 創吾  
 (72) 発明者 佐藤 浩之  
 東京都大田区下丸子3丁目30番2号キヤ  
 ノン株式会社内  
 審査官 佐伯 憲太郎

最終頁に続く

(54) 【発明の名称】情報処理装置、その制御方法、プログラム、記憶媒体

## (57) 【特許請求の範囲】

## 【請求項 1】

人が手を動かして操作を行う空間内の位置情報が反映された入力画像を取得する画像取得手段と、

前記画像取得手段によって取得された入力画像のうち前記人の手が写る部分に反映された前記位置情報に基づいて、前記人の手のうちの複数の指先のそれぞれの位置情報を検出する位置検出手段と、

前記位置検出手段によって検出された前記複数の指先の関係を使って、前記複数の指先のうち所定の対象面にタッチしたかを判定する処理の対象とする指先を特定する特定手段と、

前記特定手段によって特定された指先について前記位置検出手段によって検出された位置情報と、前記所定の対象面の位置情報とによって示される、前記特定された指先と前記所定の対象面との近接の程度に基づいて、前記指先によって前記所定の対象面上に入力されるタッチ操作を認識する認識手段と、

前記認識手段の前記近接の程度を判定するための条件として、閾値を決定する決定手段とを備え、

前記認識手段は、前記位置検出手段によって検出された前記複数の指先のそれぞれの位置情報が、前記決定手段によって決定された閾値よりも前記所定の対象面に近接していることを表す場合、前記指先によって前記所定の対象面がタッチされたと判定することを特徴とする情報処理装置。

**【請求項 2】**

前記特定手段は、前記位置検出手段によって検出された前記複数の指先の位置情報によって、最も前記所定の対象面に近いことが示される指先を特定し、前記複数の指先のうち他の指先の前記特定された指先に対する相対的な位置関係に基づいて、前記複数の指先のうちさらに前記判定する処理の対象とする指先を特定することを特徴とする請求項1に記載の情報処理装置。

**【請求項 3】**

前記特定手段は、繰り返し取得される前記入力画像から前記位置検出手段によって検出された前記複数の指先の位置情報によって、最も前記所定の対象面に近づくスピードが速いことが示される指先を特定し、前記特定された指先のスピードと他の指先が前記所定の対象面に近づくスピードの相対的な関係に基づいて、前記複数の指先のうちさらに前記判定する処理の対象とする指先を特定することを特徴とする請求項1又は2に記載の情報処理装置。

10

**【請求項 4】**

前記認識手段は、前記所定の対象面にタッチした指先が複数存在する場合、該複数の指先によって入力されるマルチタッチ操作を認識することを特徴とする請求項1乃至3のいずれか1項に記載の情報処理装置。

**【請求項 5】**

前記入力画像は、前記空間を上方から見下ろす画角で撮像された距離画像であって、前記距離画像の各画素値には、前記所定の対象面からの高さに相当する方向の位置情報が反映されることを特徴とする

20

請求項1乃至4のいずれか1項に記載の情報処理装置。

**【請求項 6】**

前記決定手段は、前記入力画像から抽出される1つの人の手に含まれる複数の指先に対して、各指先の位置情報の相対的な関係に基づいて前記条件を決定することを特徴とする請求項1乃至5のいずれか1項に記載の情報処理装置。

**【請求項 7】**

前記位置検出手段によって検出された複数の指先の種類を識別する識別手段を更に備え、

前記認識手段は、前記識別手段によって識別された特定の種類の指に対して、前記閾値を決定することを特徴とする請求項1乃至6のいずれか1項に記載の情報処理装置。

30

**【請求項 8】**

画像取得手段により、人が手を動かして操作を行う空間内の位置情報が反映された入力画像を取得する画像取得工程と、

位置検出手段により、前記取得された入力画像のうち前記人の手が写る部分に反映された前記位置情報に基づいて、前記人の手のうちの複数の指先のそれぞれの位置情報を検出する位置検出工程と、

特定手段により、前記検出された前記複数の指先の関係を使って、前記複数の指先のうち所定の対象面にタッチしたかを判定する処理の対象とする指先を特定する特定工程と、

認識手段により、前記特定された指先について前記位置検出工程で検出された位置情報と、前記所定の対象面の位置情報とによって示される、前記特定された指先と前記所定の対象面との近接の程度に基づいて、前記指先によって前記所定の対象面上に入力されるタッチ操作を認識する認識工程と、

40

決定手段により、前記近接の程度を判定するための条件として、閾値を決定する決定工程とを有し、

前記認識工程では、前記検出された前記複数の指先のそれぞれの位置情報が、前記決定された閾値よりも前記所定の対象面に近接していることを表す場合、前記指先によって前記所定の対象面がタッチされたと判定することを特徴とする情報処理装置の制御方法。

**【請求項 9】**

コンピュータを、に、請求項1に記載された情報処理装置として機能させるプログラム

50

。

### 【請求項 10】

請求項9に記載されたプログラムを格納したことを特徴とするコンピュータが読み取り可能な記憶媒体。

### 【発明の詳細な説明】

#### 【技術分野】

#### 【0001】

本発明は、画像に基づいてタッチ操作を認識する技術に関する。

#### 【背景技術】

#### 【0002】

近年、プロジェクトによる重畠表示を行う新しいユーザーインターフェース（以下、UI）システムが開発されている。上述のシステムでは、テーブルや壁、あるいは文書や物体にボタン等のタッチ操作可能なユーザーインターフェース部品（以下、UI部品）を投影し、UI部品をユーザがタッチ操作する。上述のシステムは、大型のディスプレイを備える装置に比べると小さく構成することが可能であるため持ち運びが比較的容易であり、かつ物理媒体と電子媒体を組み合わせたリッチなインタラクションが可能になるといった利点がある。上述のシステムにおいては、タッチパネルを利用せず、様々な物体に対象面を設け、対象面に対するタッチ操作を認識するために、センサやカメラ等で撮像した画像を使って、ユーザの指などの操作体が対象面に接触したことを検出することが多い。具体的には、対象面からの距離に閾値を設定し、閾値よりも対象面に近い範囲に指先が存在する状態を、タッチ中の状態とみなす。

10

#### 【0003】

特許文献1には、距離センサを用いて指先と対象物の3次元位置を取得し、指の速さや指と対象物との距離によってタッチ認識を行う方法が開示されている。また、特許文献2では、操作体のうち最も対象面に近い位置を指先位置として決定するため、手の複数の指先のうち、最も対象面に近い指先の位置を特定してその位置を指先位置とすることが開示されている。

20

#### 【先行技術文献】

#### 【特許文献】

#### 【0004】

30

【特許文献1】特開2014-157515号公報

【特許文献2】特開2014-142695号公報

#### 【発明の概要】

#### 【発明が解決しようとする課題】

#### 【0005】

ユーザが複数の指のうちの一部で対象面をタッチする際には、タッチに用いられる指の動きに付随して、タッチに用いられない他の指が対象面に近づく現象がある。タッチパネルのように直接物体が接触したことを検出するデバイスを利用する場合であれば、そのようなタッチに利用されない指の動きがタッチ検出の精度に影響することはない。しかしながら、撮像画像を用い、操作体の対象面からの距離と閾値の比較に基づいてタッチを検出する場合には、タッチに利用することが意図されない指までが、閾値よりも対象面に近い範囲に侵入してしまうことでタッチが検出されることがありうる。一方で、マルチタッチ操作と呼ばれる複数の指を利用するタッチ操作が入力される場合には、ユーザが意図を持って複数の指を同時に対象面に近づけることもある。

40

#### 【0006】

従来技術では、複数の指がタッチ対象面に近接する場合、操作に用いられない指が含まれる可能性と、マルチタッチ操作のために複数の指がタッチに用いられる可能性の両方が存在しえること、それらを区別することは考慮されていない。

#### 【0007】

本発明は上記課題に鑑みてなされたものであり、所定の対象面に対する指のタッチを両

50

者の近接の程度に基づいて検出する場合において、操作に利用される指に連動して動いてしまう他の指の影響により生じるタッチ操作の誤認識を低減することを目的とする。

【課題を解決するための手段】

【0008】

本発明は、上記課題を解決するため、人が手を動かして操作を行う空間内の位置情報が反映された入力画像を取得する画像取得手段と、前記画像取得手段によって取得された入力画像のうち前記人の手が写る部分に反映された前記位置情報に基づいて、前記人の手のうちの複数の指先のそれぞれの位置情報を検出する位置検出手段と、前記位置検出手段によって検出された前記複数の指先の関係を使って、前記複数の指先のうち所定の対象面にタッチしたかを判定する処理の対象とする指先を特定する特定手段と、前記特定手段によって特定された指先について前記位置検出手段によって検出された位置情報と、前記所定の対象面の位置情報とによって示される、前記特定された指先と前記所定の対象面との近接の程度に基づいて、前記指先によって前記所定の対象面上に入力されるタッチ操作を認識する認識手段と、前記認識手段の前記近接の程度を判定するための条件として、閾値を決定する決定手段とを備え、前記認識手段は、前記位置検出手段によって検出された前記複数の指先のそれぞれの位置情報が、前記決定手段によって決定された閾値よりも前記所定の対象面に近接していることを表す場合、前記指先によって前記所定の対象面がタッチされたと判定する。

10

【発明の効果】

【0009】

20

本発明によれば、所定の対象面に対する指のタッチを両者の近接の程度に基づいて検出する場合において、操作に利用される指に連動して動いてしまう他の指の影響により生じるタッチ操作の誤認識を低減することができる。

【図面の簡単な説明】

【0010】

【図1】情報処理装置を利用したテーブルトップシステムの外観の一例を表す図

【図2】情報処理装置を利用したテーブルトップシステムの構成の一例を示す図

【図3】タッチの認識の原理を表す図

【図4】シングルタッチを入力する手の状態と認識結果の一例を表す図

【図5】マルチタッチを入力する手の状態と認識結果の一例を表す図

30

【図6】マルチタッチを入力する手の状態と認識結果の一例を表す図

【図7】マルチタッチ認識処理の認識結果を利用するアプリケーションの一例を示す図

【図8】タッチ操作認識処理の流れの一例を示すフローチャート

【図9】シングルタッチ操作認識処理の流れの一例を示すフローチャート

【図10】各指先が操作に関わる指とみなされるかの判定処理の流れの一例を示すフローチャート

【図11】マルチタッチ認識処理の流れの一例を示すフローチャート

【図12】指同士の関係に基づく閾値の決定処理の流れの一例を示すフローチャート

【図13】平面以外の物体に対するタッチ認識の原理を示す図

【図14】タッチ判定の条件を指ごとに設定する場合の一例を示す図

40

【発明を実施するための形態】

【0011】

以下に、図面を参照して本発明の実施形態を詳細に説明する。なお、以下で説明する実施形態は、本発明を具体的に実施した場合の一例を示すものであり、これに限るものではない。

【0012】

[第1の実施形態]

<システムの外観と構成>

図1は、本実施形態で説明する情報処理装置100を設置したテーブルトップシステムの外観の一例を示している。情報処理装置100は、プロジェクタの投影光照射部105

50

から投影光を照射することにより、テーブル上や壁面など、任意の平面を対象面として設定できる。図1に示すテーブルトップシステムの場合は、情報処理装置100をテーブル面101上に設置し、テーブル面上に表示画像を投影する。ここで画像102は、プロジェクタによってテーブル面上に投影されたUI部品の1つである。以下では、プロジェクタによってテーブル面上に投影されるUI部品や写真などの各種画像などを総称して表示アイテムという。

#### 【0013】

また、受光部106は、赤外パターン投影方式（あるいはTime-of-Flight方式でもよい）の距離画像センサ115が得る距離画像の視点を示す。一例として図1のシステムの場合は、発光部107から照射された赤外光が、被写体表面で反射した反射光を受光部106で受光し、撮像する。本実施形態では、受光部106は、対象面に対して上方から見下ろす画角で撮像する位置に設置される。従って、距離画像センサ115によって得られた距離画像の各画素には、受光部106から被写体表面までの距離が反映される。なお、距離画像の取得方法は、一例として環境光やテーブル面の表示の影響が小さい赤外パターン投影方式を基に説明するが、用途に応じて視差方式や赤外光反射時間方式などを利用することも可能である。操作領域104は、本実施形態において、ユーザの操作を認識可能な範囲を表す。また操作領域104は、プロジェクタによる投影が可能な範囲、及び距離画像センサ115の視野範囲と一致する。

#### 【0014】

なお、対象面101を上方から見た画像が得られる構成であれば、必ずしも発光部107及び受光部106は上方に設置されている必要はない。例えばミラーを用いて反射光を撮像するように構成しても構わない。同様に投影光照射部105も、図1の例では、斜め上方から見下ろすように対象面101上への投影を行うが、図示されたものとは異なる方向に向けて投影された投影光を、ミラーなどを利用して対象面101に導いてもよい。本実施形態を、対象面101が鉛直方向に沿って設置されたシステムで利用される距離画像センサやプロジェクタに適用する場合でも、同様にミラーを含む光学系を利用可能である。

#### 【0015】

本実施形態では、ユーザが手103aや手103bなどに示すように、複数の方向から手を、操作領域104上の操作空間（対象面101に設定された操作領域104表面と距離画像センサ115の受光部106の間の空間）に挿入可能である。ユーザが手を使って、テーブルトップシステムに対して、表示アイテムを操作対象とするジェスチャ操作を入力する場合の情報処理装置100の動作について述べる。ただし、本実施形態は、表示アイテムがテーブル面101に投影される場合だけでなく、例えばホワイトボードや任意の壁面に投影光が照射される場合や、投影面が平面でない場合であっても適用可能である。

#### 【0016】

本実施形態では、図1に示されるように操作領域104に対して平行な二次元平面にx軸とy軸、操作領域104に直交する高さ方向にz軸を設定し三次元位置情報を座標値として扱う。ただし、対象面が平面でない場合やユーザと対象面との位置関係によっては、必ずしも対象面に平行あるいは直交の関係の座標軸ではなくてもよい。その場合も、z軸は認識対象と対象面との近接の程度（両者の間の距離の大きさの程度）を検出するため対象面と交差する方向に設定され、x軸とy軸はz軸と交わる方向に設定される。本実施形態では、入力された距離画像の各画素値に対して、受光部106のレンズ特性および対象面101との相対位置関係に基づく座標変換を施す。本実施形態において取得される距離画像の各画素には、受光部106から被写体表面までの距離の大きさに対応する値が、画素値として保持されている。従って、座標変換により、距離画像内の各画素の座標がテーブルに定義された実世界上の座標系にマッピングされ、x、y、z座標の値を得ることができる。

#### 【0017】

なお、本実施形態では、ユーザの指先がタッチ対象面に触れたか否かを判定するために

10

20

30

40

50

検出する、指先とタッチ対象面の近接の程度は、距離画像の画素値に基づいて得られる距離情報を例に説明する。ただし、指先とタッチ対象面の近接の程度として利用可能な情報はこれに限らない。例えば、操作体がタッチ対象面に近接することによって生じる静電容量の変化量や、温度や圧力、接触面積の変化量の情報であってもよい。

【0018】

図2(A)は、本実施形態における情報処理装置100のハードウェア構成の一例を示すブロック図である。同図において、CPU110は、バス113を介して接続する各デバイスを統括的に制御する。オペレーティングシステム(OS)をはじめ、後述するフローチャートに示されるような本発明に係る各処理プログラム、デバイスドライバ等はROM112に記憶されており、RAM111に一時記憶され、CPU110によって適宜実行される。RAM111は、高速にアクセス可能なCPU110の主メモリ、ワークエリア等の一時記憶領域として用いられる。ここでOSおよび各処理プログラム等は外部の記憶装置116に記憶されていてもよく、その場合は電源投入時に必要な情報がRAM111に適宜読み込まれる。また、ディスプレイI/F117は、情報処理装置100内部で生成される表示アイテム(表示画像)をプロジェクトタ118が処理可能な信号に変換する。入出力I/F114は、距離画像センサ115から距離情報を取得し、情報処理装置100が処理可能な情報に変換するとともに、記憶装置116と情報処理装置100間で相互にデータの変換を行う。

【0019】

本実施形態では、情報処理装置100で投影するデジタルデータは記憶装置116に格納されているものとする。記憶装置116としては、ディスクデバイスやフラッシュメモリ、ネットワークやUSBなどの各種の入出力I/F114を介して接続される記憶装置116を使用する。本実施形態において、距離画像センサ115は、操作領域104上の情報を取得するために用いられる撮像部である。距離画像センサ115によって取得された画像は入力画像としてRAM111で一時保存され、CPU110によって適宜処理され、破棄されるが、適宜必要なデータは記憶装置116に蓄積しても構わない。

【0020】

なお図2(A)の例では、距離画像センサ115、プロジェクトタ118はそれぞれ情報処理装置100に一体化したデバイスである。しかしながら、入出力用のインターフェースを介して接続された外部装置であり、情報処理装置100と協働して情報処理システムを構成するようにしてもよい。また、さらに操作領域104内に載置されたドキュメントや立体物の可視光画像を撮像する可視光カメラ等を加えて、書画カメラとしての機能を有するシステムとして構成することもできる。

【0021】

図2(B)は、本実施形態における情報処理装置100の機能構成の一例を示すブロック図である。情報処理装置100は、少なくとも画像取得部120、領域検出部121、位置検出部122、特定部123、条件決定部124、識別部125、認識部126から構成される。これらの各機能部は、CPU110が、ROM112に格納されたプログラムをRAM111に展開し、後述する各フローチャートに従った処理を実行することで実現されている。また例えば、CPU110を用いたソフトウェア処理の代替としてハードウェアを構成する場合には、ここで説明する各機能部の処理に対応させた演算部や回路を構成すればよい。また保持部126は、ROM112あるいは、記憶装置116のいずれかに対応する機能部であり、後述する連動指先補正に用いるルックアップテーブルや、プロジェクトタ118に出力される画像の基となる画像データ等を保持する。

【0022】

画像取得部120は、距離画像センサ115によって撮像された距離画像を、入力画像として一定時間毎に取得し、RAM111に随時保持する。なお画像取得部120が取得し、各機能部とやりとりする対象は、実際には画像データに対応する信号であるが、本明細書では単に「距離画像を取得する」あるいは「入力画像を取得する」として説明する。

【0023】

10

20

30

40

50

領域検出部 121 は、画像取得部 120 によって取得された入力画像の各画素について、閾値判定やノイズ低減処理を施し、距離画像から 1 以上の手領域を抽出する。手領域とは、入力された距離画像のうち、操作体としてユーザが利用する手が写っている領域である。本実施形態の場合は距離画像の画素値によって、対象面 101 よりも高い位置に存在することが示される被写体が写る領域で、かつ、画像端に接している領域を手領域として抽出する。

#### 【0024】

位置検出部 122 は、領域検出部 121 によって抽出された手領域の輪郭の情報に基づき、ユーザの手の 1 以上の指を検出し、その先端とみなした 1 点の 3 次元位置を表す座標値を、指先の位置情報として取得する。本実施形態で入力画像として扱う距離画像の場合、指先に相当する画素の画素値は、ユーザの指先の上面で反射された赤外光の輝度値に相当する。この輝度値は、発光部 107 から照射された赤外光の反射光強度を表すため、強度が強いほど距離画像センサ 115 からユーザの指の上面までの距離が近く、強度が弱いほど距離画像センサ 115 から指までの距離が離れているとみなせる。このように本実施形態の場合は、距離画像の中で特定された指先位置の画素値が、距離画像センサ 115 と指先の距離を反映した距離情報として扱われる。そのため情報処理装置 100 は、操作空間内での指先位置の位置情報（画像内の画素を示す位置情報）及び画素値に対し、所定のキャリブレーションパラメータを反映させて、図 1 に示されている（x、y、z）座標に変換し、指先の 3 次元位置として取得する。指先の z 方向の座標値は「指先高さ」ともいう。本実施形態では指先高さの尺度として三次元空間上のユークリッド距離（mm）を用いるが、別の距離尺度を用いてもよい。

10

#### 【0025】

特定部 123 は、位置検出部 122 が検出した指先が複数であった場合、複数の指先のそれぞれの位置情報の相対的な関係に基づいて、タッチ操作の入力に用いられるとみなされる指を特定する。本実施形態では、最もタッチ操作の入力に用いられる可能性が高いとみなされる指先を 1 つ特定した上で、特定された指先に対する相対的な関係に基づいて、他の指先を、タッチ操作の入力に用いられる指とみなすか判定する条件を決定する。そして決定された条件を使って、他の指を絞り込むことにより、全体からタッチ操作の入力に用いられるとみなされる指を特定する。特定された指先は、後述する認識部 126 によって行われる、タッチ対象面にタッチしたかを判定する処理の対象となる。

20

#### 【0026】

条件決定部 124 は、位置検出部 122 が検出した指先それぞれの指先に対して対象面であるテーブル 101 にタッチしたかを判定するための条件を決定する。また本実施形態では、条件決定部 124 はさらに、テーブル 101 から指先が意図的に離された、つまりはタッチ入力が終了されたことを判定するための条件を決定する。本実施形態では、タッチ対象面と指先の間の距離がと閾値の比較によって、タッチを検出する。従って条件決定部 124 は、上記の条件として閾値の大きさを決定する。

30

#### 【0027】

識別部 125 は、検出された指先の幾何学的な位置関係に基づいて、腕の左右を識別する。例えば、指先が 5 つ検出され、侵入位置から見て、最も左側にある指先の座標が、他のどの指先よりも侵入位置の座標に近い場合には、最も短い指（親指）が左側に位置することになるので、その腕領域は右手に相当すると認識される。ただし、腕の左右の認識方法はこれに限られず、腕形状のパターンを学習した SVM (Support Vector Machine) を用いる等してもよい。さらに識別部 125 は、腕の左右の識別結果と指先の幾何学的な位置関係に基づいて、検出された複数の指先の種類を識別する。例えば、右手の指先が 5 つ検出された場合には、侵入位置から見て、時計回りに、小指、薬指、中指、人差し指、親指であると識別する。

40

#### 【0028】

認識部 126 は、位置検出部 122 が検出したユーザの各指先の位置情報と、条件決定部 124 が決定した条件を使って、指先が前記所定の対象面にタッチしたことを認識する

50

。本実施形態では、認識部 126 が認識可能なタッチ操作にはシングルタッチ操作とマルチタッチ操作が含まれる。認識部 126 がタッチを認識する処理の具体的な内容は図 3 を参照して後述する。

#### 【0029】

出力制御部 127 は、ROM 112 や記憶装置 116 に記憶された情報を用いて、本実施形態の表示部であるプロジェクト 118 によって、テーブル 101 上に投影させる画像を生成する。例えば、マルチタッチ操作の認識結果に基づいて、表示中の画像の少なくとも一部に対して、拡大／縮小、回転などの変形を施し、表示部に出力させる。出力された画像は、プロジェクト 118 によって、対象面であるテーブル 101 上に投影される。投影される画像は、複数の表示アイテムを含み、アイテム毎に移動や拡大／縮小、回転が可能であってもよく、画像全体が移動や拡大／縮小、回転の対象であってもよい。

10

#### 【0030】

この他、手ぶり等の空間ジェスチャ動作を認識するジェスチャ認識部、プロジェクト 118 に出力される画像を制御する表示制御部等、情報処理装置 100 の使用目的やアプリケーションに応じた機能部を構成することができる。なお上述した各機能部は、RAM 111 を、本実施形態に係る処理で利用される情報を一時的に保持する保持部として利用する。ただし、保持部は ROM 112、記憶装置 116 等の一部であっても構わない。

#### 【0031】

以下、ユーザが操作の入力に用いる操作体及びその先端の一例として、ユーザの手 103a、b 及びその指を挙げて第 1 の実施形態を説明していく。ただし、本実施形態では操作体として、生身の手指に限らず、義手やロボットアームなどを利用し、指に相当する部位を識別する必要がある場合にも適用可能である。

20

#### 【0032】

##### <タッチの認識処理>

ここで、認識部 126 が実行するタッチ認識処理の概要を説明する。認識部 126 は、検出された各指先について、次の二つの状態を検出する。一つはタッチ状態である。タッチ状態は、指先が指先の指先高さがタッチ閾値を下回っている場合であって、指先と対象面がタッチしているとみなされる程度に十分近接している状態である。十分近接しているか否かを判定するための条件は、条件決定部 124 によって指毎に決定される。もう一つの状態は非タッチ状態である。非タッチ状態は、ユーザが意図的に指先を対象面から離している状態であり、指先高さがリリース閾値を上回っている場合に認識される。本実施形態では、リリース閾値もまた、条件決定部 124 によって指毎に決定される。

30

#### 【0033】

本実施形態では、タッチ閾値とリリース閾値としては、それぞれ異なる値が用いられ、特に、タッチ閾値よりリリース閾値が大きい値である。これは、指先を近接させる時と離す時とで指先の動き方が異なることや、センサに検出誤差が発生することを踏まえた上で、タッチ状態の間は対象面から指先が十分に離れるまでリリースが検出されないようにするための工夫である。これにより、検出誤差や指先のぶれにより、タッチ操作の継続中に誤ってリリースが検出されてしまうという誤認識が低減される。ただし、タッチ閾値とリリース閾値の差分は、センサの検出精度や装置の設置環境に応じて調節されればよく、両者は一致しても構わない。また、多くの場合、タッチ閾値とリリース閾値の大きさは、指の厚みに対して十分に大きく設定される。これは、距離画像センサから得られる距離情報にノイズがある場合でもロバストなタッチ認識を可能にするためである。ただしこの点も、センサの検出精度や装置の設置環境に応じて調節されればよい。

40

#### 【0034】

以降、本実施形態において「タッチの認識」とは、指先がタッチ閾値より小さいか、あるいは指先がリリース閾値より大きくなかったかを判定することにより、指先の状態を判定することをいう。それに対し、「タッチ操作の認識」とは、タッチ状態での指先の位置や移動軌跡が辞書に記憶された各種コマンドに対応するものであった場合に、情報処理装置 100 に入力された指示として解釈することをいう。本実施形態においてタッチ操作とい

50

う語は、指先高さの変動によって、タッチ状態および非タッチ状態の相互の遷移が発生する、指先による操作一般を指す。特に、シングルタッチ操作という語は、1つの指先によるタッチ操作一般を指す。本実施形態ではシングルタッチ操作の一例としてタップ操作を挙げて説明する。タップ操作とは、一般的に、対象面上のある位置をタッチした後、スライド移動させることなく所定時間内にリリースする操作を指し、主に対象面上のアイテムを選択する場合に利用される。シングルタッチ操作に対し、マルチタッチ操作という語は、複数の指先によって1つの指示が入力されるタッチ操作一般を指す。本実施形態では、マルチタッチ操作の例として、2箇所のタッチ位置の間隔を広げたり狭めたりすることによって表示画像の拡大縮小を指示するピンチ操作と、2箇所のタッチ位置を相対的に回転させることによって表示画像の回転を指示する回転操作を挙げる。

10

#### 【0035】

本実施形態では、認識部126は、指先が非タッチ状態からタッチ状態に遷移した場合、TOUCHイベントをアプリケーションに通知する。また、タッチ状態から非タッチ状態に遷移した場合は、RELEASEイベントを発行する。以後、認識部126が、後に続く処理のトリガーとなるイベントを出力制御部127やアプリケーションに通知することを、「イベントを発行する」と表現する。イベント情報は、どの指先が、どの位置で状態遷移を行ったかという情報とともに通知される。通知を受けたアプリケーションは出力制御部127に対し、イベント情報と指先の情報及びUI部品の情報に基づいて、投影される画像を更新させる。

#### 【0036】

ここで図3(A)～(C)を用いて、タッチに伴う指先高さの変動、上述した指先の状態遷移、イベントの発行について具体的に説明する。図3(A)に指202の指先203のタッチを行う前の様子を示す。指先高さ204はリリース閾値201を上回っているため、指先203は非タッチ状態であると認識される。状態遷移が発生していないため、イベントは発行されない。図3(B)は、指先が対象面(Z=0の面)をタッチした瞬間の様子である。指先高さ205はタッチ閾値200を下回っているため、指先203はタッチ状態であると認識される。非タッチ状態からタッチ状態へと指先203の状態が遷移したため、TOUCHイベントが発行される。図3(C)はタッチしてすぐに(タップ操作に定義された閾値時間の間に)、指先が対象面から離された様子を表す。指先高さ206はリリース閾値201を上回っているので、指先203は非タッチ状態であると認識される。タッチ状態から非タッチ状態への指先203の状態が遷移したことで、RELEASEイベントが発行される。このとき、タッチ状態から非タッチ状態へと遷移するまでの時間がタップ時間閾値以内、かつタッチ状態になった瞬間のx y座標と非タッチ状態になった瞬間のx y座標のずれがタップずれ閾値以内であったとする。タップ操作が認識される条件が満たされたことで、タップ操作が入力されたことを示すTAPイベントが発行される。

20

#### 【0037】

認識部126におけるタッチ認識処理は、大きく分けてシングルタッチ認識処理とマルチタッチ認識処理に分かれている。シングルタッチ認識処理はある1つの指先のタッチ状態であるか非タッチ状態であるかの認識を行い、TOUCHイベント、RELEASEイベント、TAPイベントを発行する。マルチタッチ認識処理では、前段階として実行されるシングルタッチ認識処理の結果と、各指先の位置関係に基づいてタッチ状態にある複数の指先のペアリングが行われる。そして、ペアリングされた指先の位置の変化によってピンチ操作およびローテート操作が認識され、それらが入力されたことを表すPINCHイベントおよびROTATEイベントが発行される。

30

#### 【0038】

本実施形態は、上述したようなタッチ認識において、ユーザが操作に用いる意図を持って対象面に近づけた指に運動して、操作に用いられない指が動いた結果タッチ閾値よりも低い高さで指先がされてしまい、タッチが誤検出されることを課題として認識する。そこで、本実施形態では、複数の指先を検出した場合、操作に用いられる可能性が高いとみな

40

50

される指を絞り込み、絞り込まれた指を対象にタッチの認識を行う。言い換えると、操作に用いられない指であるにも関わらず他の指に連動して対象面に近づいてしまった指を、タッチ認識処理の対象としないことで、タッチの誤認識を低減する。

【0039】

＜タッチ操作認識処理の流れ＞

図8のフローチャートを参照して、本実施形態のメイン処理となるタッチ操作の認識処理の流れを説明する。本実施形態では、図8のフローチャートの処理は、距離画像センサ115から距離画像が入力される毎に周期的に繰り返される。従って処理が繰り返される周期は、距離画像センサ115の撮像映像のフレームレートに一致する。

【0040】

ステップS100において、画像取得部120は距離画像センサ115によって撮像された距離画像を、入力画像として一定時間毎に取得し、RAM111に随時保持する。距離画像センサ115によって撮像された距離画像の画素値は距離画像センサ115までの距離を反映している。ステップS101において、領域検出部121は、画像取得部120によって取得された距離画像から人の腕が写る領域（腕領域）を抽出し、その輪郭を示す位置情報を取得し、RAM111にその情報を保持する。

【0041】

ここで人の「腕」とは、人の肩から指先にかけての部位の全てを指し、撮像画像のうち、人の腕に該当する部分が一部写る領域が、腕領域である。操作体または本実施形態において「手」とは、腕のうち、手首から先の全てを差す。「手」には、5指、掌、甲が含まれる。領域検出部121の実際の処理としては、取得した距離画像の各画素が示すz方向の座標値に対して閾値処理を行うことで、テーブルよりも高さが高い座標値を持つ画素群からなる領域であって、かつ画像端と接触する領域を、腕領域として抽出する。図1において、腕領域は、手103aおよび手103bのうち、操作領域104に侵入している部分（破線で表された操作領域104の境界より内側の部分）が、撮像画像に写った部分に相当する。ただし、腕領域の抽出方法はこれに限らず、例えば別途撮像された操作領域104のRGB画像における肌色領域に対応する部分を抽出する等してもよい。また、本実施形態の領域検出部121は、腕領域が抽出された入力画像に微分フィルタを適用することに基づいて、輪郭線の座標を取得する。さらに、領域検出部121は腕領域と画像端が接触するピクセルの平均位置（以下では、操作領域に対する腕領域の侵入位置と称する）を算出し、これをRAM111に保持する。

【0042】

ステップS102において、位置検出部122は、取得された腕領域から、1以上の指先位置を検出する。本実施形態では、位置検出部122はまず、手領域の輪郭線が領域の内側になす角度に基づいて指の先端とみなす1画素を特定する。ただし、指の先端の特定方法は輪郭が成す内角を基準とする方法に限られず、例えば輪郭の極値を算出する等してもよい。さらに位置検出部122は検出した指先の画素の位置および画素値の情報から、指先の位置情報として高さを含む3次元位置を取得し、RAM111にその情報を保持する。ステップS103において、識別部125は、検出された指先の幾何学的な位置関係に基づいて、腕の左右と指先の種類を識別する。腕と指先の識別結果はRAM111に保持される。また、種類認識部124は腕と指先のトラッキングを行う。そして現フレームの腕や指先の種類を認識することが困難な場合には、前フレームまでの認識結果を用いて、現フレームの腕と指先の認識情報を補完する。ステップS104において、条件決定部124は、前フレームと現フレームの指先高さの変動から指先の上下移動のスピードを算出する。

【0043】

ステップS105において、条件決定部124は、検出されている各指のタッチ認識条件を決定する。タッチ認識条件とは、指先がタッチ対象面をタッチしたかを判定するための条件である。第1の実施形態の場合、予め設定されているタッチ閾値及びリリース閾値の値をROM112や記憶装置116から読み出すことで、条件として決定する。

10

20

30

40

50

## 【0044】

ステップS106において、認識部126は、シングルタッチ認識処理を行う。本実施形態では、シングルタッチ認識処理の中で、ユーザがタッチ操作に用いる可能性が高い指を絞り込んだ上で、タッチ認識を行う。シングルタッチ操作認識処理の詳細は後述する。ステップS107において、認識部126は、マルチタッチ操作処理を行う。本実施形態では、ステップS106の処理でタッチ状態にあると認識された指が、1つの手につき複数存在した場合は、それらをペアリングし、ステップS107において、マルチタッチ操作の認識を行う。ただし、ペアリングされた指先が、マルチタッチ操作としての意味を持つ動作を成していなかった場合は、ステップS107では認識結果を出力せず、それぞれをシングルタッチ操作として認識する場合がある。マルチタッチ操作認識処理の詳細は後述する。ステップS108では、出力制御部127が、ステップS106あるいはステップS107で認識されたタッチ操作に対する応答を出力する。例えば、プロジェクト118によってテーブル101に投影する画像を生成し、出力する。以上が、本実施形態におけるメインの処理の流れである。

10

## 【0045】

<指先が操作に関わる指とみなされるかの判定>

ここで、本実施形態で実行される指先が操作に関わる指とみなされるかの判定方法について説明する。本実施形態では、ステップS106において実行されるシングルタッチ認識処理の中で、ユーザがタッチ操作に用いる可能性が高い指を絞り込む処理を行う。まずは、図4(A)～(C)を参照して、本実施形態で実行されるこの絞り込みについて説明する。

20

## 【0046】

図4(A)は、ユーザが右手を、指を曲げたり伸ばしたりする意思を持たずに、自然な姿勢で維持している状態でテーブル面101上に静止させた状態を、y軸方向から見た図である。以下では、このようなユーザの手の自然な状態をニュートラルポジションという。図中の5つの円は指先の位置情報が検出された5つの指先を示している。タッチを認識するためにタッチ閾値と比較される指先の高さは、円の上面かつx軸と平行な接線のz座標である。図4(A)では、中指302が最も指先高さが高く、次いで薬指301や人差し指303の高さが高く、小指300と親指304は最も指先高さが低い。

30

## 【0047】

次に、ユーザの右手の状態が、ニュートラルポジションから変移し、人差し指303でタッチ対象面(  $Z = 0$  の面)に対して、タッチを行う場合を考える。まず図4(B)は、全ての指に対して、図3で説明した一定のタッチ閾値200とリリース閾値201を適用してタッチ認識を行う場合を表す。全ての指に一定のタッチ閾値200を適用した場合に、タッチ状態であると認識される指先が、黒塗りされた円で表される。図4(B)の場合、ユーザは人差し指303でタッチをしたつもりであるにも関わらず、小指300と親指304もタッチ閾値を下回る。この場合、人差し指303に加えて、小指300と親指304もタッチ状態であると認識され、各指先についてTOUCHイベントが発行される。こうした意図しない指先のタッチの誤認識は、意図しない場所のUI部品が反応してしまったり、シングルタッチ操作の入力が意図されているにも関わらず、マルチタッチ操作が認識されたりする原因にもなりえる。

40

## 【0048】

ここで図4(C)は、本実施形態によって、指先の相対的な位置関係に基づいて、タッチ閾値200と比較する指先を絞り込み、タッチ対象面にタッチしているかを判定する場合を表す。図4(C)に示す手及び各指先の姿勢は、図4(B)と同一である。本実施形態では、検出されている複数の指先のうち、最も高さが小さい指を特定し、特定された指との相対的な高さを使って、他の指先の中でタッチ閾値200によるタッチ判定処理の対象とする指を絞り込む。本実施形態では、最も高さが小さい指はタッチ操作に用いられる可能性が最も高い指であるとみなすため、その高さを基準として効率の良い絞り込みを行うことができる。図4(C)の場合、検出されている複数の指先の中で、最も高さ

50

が小さい指先は、人差し指 3 0 3 であると特定される。例えば、特定された人差し指 3 0 3 の高さを、 $H_{min}$ とする。そして、検出されている複数の指先の中で、指先の高さが、絞り込み閾値 3 0 5 を表す  $H_{min} + H$  ( $H$  は正の定数) を下回るか否かを判定し、絞り込み閾値よりも高さが小さい指先を絞り込む。絞り込み閾値 3 0 5 は、最も高さが小さい指先を基準とする相対的な高さの閾値である。本実施形態では、このように絞り込まれた指先が、ユーザがタッチ操作の入力に用いる意図を持っている指である可能性が高いとみなす。一方で、絞り込み結果に含まれていない指先は、操作の入力に用いることが意図されていない指先であるとみなす。そして本実施形態では、絞り込まれた指先の高さが、タッチ閾値 2 0 0 を下回るか否かに基づいて、実際にタッチ入力が行われているかを判定する。

## 【0049】

10

図 4 (C) の例では、人差し指 3 0 3 の指先高さは  $H_{min}$  であり、 $H_{min} + H$  を下回っているのでタッチ閾値 2 0 0 との比較によるタッチ認識処理の対象となる。そして人差し指 3 0 3 の指先高さ  $H_{min}$  はタッチ閾値 2 0 0 を下回っているので、人差し指 3 0 3 はタッチ状態であると認識される。人差し指以外の指は指先高さが  $H_{min} + H$  を上回っているので、タッチ閾値 2 0 0 によるタッチ判定の対象から外れ、タッチ状態であると認識されることはない。このようにして、図 4 (B) のように誤認識抑制を行わない場合にタッチ状態になってしまう小指 3 0 0 と親指 3 0 4 の誤認識を防ぐことができる。

## 【0050】

なお、上述したようにタッチ認識処理の対象とする指先を絞り込むことは、タッチ閾値 2 0 0 を予め小さい値に設定することと等価ではない。タッチ閾値 2 0 0 は、センサの精度や環境によって生じてしまう位置情報の検出誤差やぶれを許容できるように、ある程度大きく設定される必要があるからである。タッチ閾値を小さくすると、検出誤差やぶれにより、ユーザがタッチしたつもりなのに情報処理装置 1 0 0 がそれを認識できないといった問題を発生させる可能性がある。従って本実施形態では、十分な大きさを持ったタッチ閾値 2 0 0 を設定し、タッチ閾値 2 0 0 と比較する指先を絞り込むことによって、ユーザがタッチしようとしている場合の認識漏れを防いでいる。

20

## 【0051】

次に、指先の移動スピードの比較によって絞り込みを行う方法について、図 5 (A) ~ (C) を参照して説明する。図 5 (A) では前フレームの指先位置を破線の円で、現フレームの指先位置を実線の円で示している。さらに、前フレームと現フレームの指先高さ位置を繋いだ矢印は、Z 軸方向のスピードの大きさを示している。全ての指先の中で最も指先スピードが大きい指先は人差し指 3 0 3 であり、その指先スピードを  $V_{max}$  とする。指先スピードが絞り込み閾値を表す  $V_{max} - v$  ( $v$  は正の定数) を上回った相対的な指先スピードが大きい指先のが、タッチ操作に用いられる指先である可能性が高いとみなされ、タッチ閾値 2 0 0 によるタッチ認識処理の対象となる。図 5 (A) では、人差し指 3 0 3 の指先スピードは  $V_{max}$  であり、 $V_{max} - v$  を上回っているのでタッチ閾値 2 0 0 によるタッチ判定が行われる。そして、人差し指 3 0 3 の指先高さはタッチ閾値 2 0 0 を下回っているので、人差し指 3 0 3 はタッチ状態であると認識される。人差し指以外の指は指先スピードが  $V_{max} - v$  を下回っているので、タッチ閾値 2 0 0 によるタッチ判定の対象から外れ、タッチ状態であると認識されることはない。

30

## 【0052】

40

本実施形態では、上述した、指先の高さを基準とした絞り込みと、指先のスピードを基準とした絞り込みの両方を段階的に行うことで精度のよい絞り込み結果を得ることができる。ただし、これらはそれぞれ効果を有するものであるため、システムの構成や状況によっては片方を省略してもよい。また、両方の処理を、結果の OR をとする形で併用してもよい。

## 【0053】

## &lt;シングルタッチ操作の認識処理&gt;

次に、本実施形態のステップ S 1 0 6 の処理で実行されるシングルタッチ操作の認識処理を、図 9 のフローチャートを参照して説明する。なお、図 9 のフローチャートの処理が

50

開始されると、ステップ S 102 で位置情報が検出された全ての指先のそれぞれに対し、後述するステップ S 201 ~ S 214 の処理を繰り返す。全ての指先に対する処理が終了した段階で、シングルタッチ認識処理は終了となり、図 8 のフローチャートに戻る。

#### 【 0054 】

ステップ S 201 において、特定部 123 は、その時点で処理対象としている指先が、タッチ操作に用いられる指とみなされるかを判定する。ステップ S 201 の処理が、図 4 を使って説明した絞り込みに相当する。なおステップ S 201 での判定は、ユーザが、処理対象としている指先をタッチ操作の入力に用いる意図を持っている可能性が高いことを、センサを使って検出している情報に基づいて推定するものである。従って判定結果は、実際のユーザの思考や意図には依存しない。

10

#### 【 0055 】

ここで、図 10 のフローチャートは、ステップ S 201 で実行される処理の流れを表すものである。ステップ S 300 において、特定部 123 は、指先高さと  $H_{min} + H$  を比較する。もし指先高さのほうが小さければステップ S 301 に進み、そうでなければステップ S 303 に進む。ステップ S 301 において、特定部 123 は指先スピードと  $V_{max} - v$  を比較する。もし指先スピードの方が大きければステップ S 302 に進み、そうでなければステップ S 303 に進む。ステップ S 302 ではタッチ操作に用いられる指であるとの判定が行われ、ステップ S 201 の判定結果は「YES」となる。その結果、この時処理対象とされている指先は、高さをタッチ閾値と比較することによりタッチ状態か否かを判定する処理の対象となる。ステップ S 303 では、タッチ操作に用いられる指でないと判定が行われ、ステップ S 201 の判定結果は「NO」となる。ステップ S 302 でタッチ操作に用いられる指であると判定された場合（ステップ S 201 で YES）、ステップ S 202 に進む。ステップ S 303 でタッチ操作に用いられる指でないと判定された場合（ステップ S 201 で NO）、処理はステップ S 208 に進む。

20

#### 【 0056 】

ステップ S 202 では、認識部 126 は、ステップ S 102 で検出された指先の位置情報のうち z 座標（高さ）が、ステップ S 105 で決定されたタッチ閾値より小さいかを判定する。指先の高さがタッチ閾値より小さいと判定された場合（ステップ S 202 で YES）、ステップ S 203 に進む。指先の高さがタッチ閾値より小さいと判定されない場合（ステップ S 202 で NO）、ステップ S 209 に進む。

30

#### 【 0057 】

ステップ S 203 において認識部 126 は、処理対象の指先が、この時点で既にタッチ状態であるかの判定を行う。タッチ状態であれば（ステップ S 203 で YES）、ステップ S 207 に進む。タッチ状態ではない、すなわち非タッチ状態であれば（ステップ S 203 で NO）、ステップ S 204 に進む。ステップ S 204 において、認識部 126 は、処理対象の指先のタッチ状態への状態遷移があったことを確定する。本実施形態では、RAM 111 に保持するタッチフラグの値を「TRUE」とし、ステップ S 205 に進む。

#### 【 0058 】

ステップ S 205 で、認識部 126 は、TOUCH イベントを発行する。ステップ S 206 において、認識部 126 は、ステップ S 204 でタッチ状態への遷移が確定した時刻の情報を、タッチ時刻として RAM 111 に保持する。ステップ S 207 において、認識部 126 は、処理対象の指先位置の位置情報を、この時点でのタッチ位置として RAM 111 に保持する。そして検出されている全ての指先が処理されるまで、処理を繰り返す。

40

#### 【 0059 】

ステップ S 208 において、認識部 126 は、処理対象の指先が、この時点でタッチ状態であるかの判定を行う。タッチ状態であれば（ステップ S 208 で YES）、ステップ S 209 に進む。タッチ状態ではない、すなわち非タッチ状態であれば（ステップ S 208 で NO）、図 9 のフローチャートの処理を終了する。あるいは、処理対象の指を変えて処理を繰り返す。

#### 【 0060 】

50

ステップ S 2 0 9において、認識部 1 2 6 は、指先高さがリリース閾値より大きいかの判定を行う。指先高さがリリース閾値より大きいと判定される場合（ステップ S 2 0 9 で YES）、ステップ S 2 1 0 に進む。指先高さがリリース閾値より大きいと判定されない場合（ステップ S 2 0 9 で NO）、ステップ S 2 0 7 に進む。ステップ S 2 1 0 において、認識部 1 2 6 は、対象の指先のタッチ状態から非タッチ状態への遷移を確定させ、RAM 1 1 1 に保持するタッチフラグの値を「FALSE」とする。ステップ S 2 1 1 において、認識部 1 2 6 は、RELEASE イベントを発行する。ステップ S 2 1 2 において、認識部 1 2 6 は、ステップ S 2 1 0 で非タッチ状態への遷移が確定した時刻の情報を、リリース時刻として RAM 1 1 1 に保持する。ステップ S 2 1 3 において、認識部 1 2 6 は、処理対象である指先について、タッチイベントが発行されてからリリースイベントが発行されるまでに得られた情報によって、タップ操作の認識条件が満たされたかを判定する。具体的には、以下の式 1 の条件が満たされるかの判定を行う。  
10

（リリース時刻 - タッチ時刻） <  $T$  かつ

（リリース位置とタッチ位置の距離） <  $D$  ( $T$ 、 $D$  は正の定数) ... (式 1)

タップ操作の条件が満たされると判定された場合（ステップ S 2 1 3 で YES）、ステップ S 2 1 4 に進む。タップ操作の条件が満たされると判定されない場合（ステップ S 2 1 3 で NO）、図 9 のフローチャートの処理を終了する。あるいは、処理対象の指を変えて処理を繰り返す。全ての指について処理を行われた段階で、シングルタッチ認識処理は終了する。

#### 【0061】

20

##### <マルチタッチ操作の認識処理>

次に、本実施形態のステップ S 1 0 7 の処理で実行されるマルチタッチ操作の認識処理を、図 1 1 のフローチャートを参照して説明する。なお本実施形態では、マルチタッチ操作の認識処理の前にシングルタッチ認識処理が行われ、各指先の状態が得られている。

#### 【0062】

30

ステップ S 4 0 0 において、認識部 1 2 6 は、タッチ状態にある複数の指先をペアリングする。本実施形態では、タッチ時刻が最も早い指先から、タッチ時刻に沿って順次 2 つの指先をペアリングする。ただしこれは一例である、他のペアリング方法を用いてもよい。なお、ステップ S 4 0 0 において複数の組み合わせがペアリングされた場合、以下のステップ S 4 0 1 ~ ステップ S 4 0 6 の処理は全ての組み合わせのそれぞれを対象に、複数回繰り返される。

#### 【0063】

ステップ S 4 0 1 において、認識部 1 2 6 は、ペアリングされている 2 つの指先の間の距離を取得する。本実施形態では、x y 平面上の二次元のユークリッド距離を用いるが、別の距離尺度を用いてもよい。ステップ S 4 0 2 において、認識部 1 2 6 は、取得した指先間の距離と、前フレームで取得された距離に差が生じているかを判定する。ここでは、フレーム間での距離の差分が、所定の値より大きいか否かで判定を行う。距離に差があると判定される場合（ステップ S 4 0 2 で YES）、ステップ S 4 0 3 に進む。距離に差があると判定されない場合（ステップ S 4 0 2 で NO）、ステップ S 4 0 4 に進む。

#### 【0064】

40

ステップ S 4 0 3 において、認識部 1 2 6 は、PINCH イベントを発行する。PINCH イベントは、2 つの指先間の距離が変化したことで拡大・縮小操作を認識した場合に通知されるイベントである。PINCH イベントは、2 つの指先間の距離の変化から計算された、拡大・縮小の割合や差分距離といった情報を伴って出力制御部 1 2 7 に通知される。例えば、拡大・縮小の割合（%）は、式 2 で計算できる。

（現在の指先間の距離） / （ペアリングされた時点での指先間の距離） × 1 0 0 ... (式 2)

また、例えば、差分距離は（現在の指先間の距離 - ペアリングされた時点での指先間の距離）で計算できる。出力制御部 1 2 7 は、例えば、PINCH イベントの通知を受け、表示アイテムの大きさを拡大したり縮小したりできる。

50

## 【0065】

ステップS404において、認識部126は、ペアリングされている指先間を結ぶ線分の角度を取得する。角度の取得方法の例は後述する。ステップS405において、認識部126は、算出した角度が前フレームと比較して変化しているかどうかを判定する。角度が変化していると判定される場合(ステップS405でYES)には、ステップS406へ進む。角度が変化していると判定されない場合(ステップS405でNO)には、処理を終了する。あるいはステップS401に戻って繰り返す。

## 【0066】

ステップS406において、認識部126は、ROTATEイベントを発行する。ROTATEイベントは、ペアリングされた2つの指先の絶対的な角度が変化したことで回転操作を認識した場合に通知されるイベントである。絶対的な角度の計算方法は、例えば、2つの指先を繋いでできる線分とx軸のなす角度を用い、時計回りの角度をプラスとし、反時計回りの角度をマイナスとすることができる。ROTATEイベントは2つの指先の絶対的な角度の変化から計算された、回転の割合等の情報を伴って出力制御部127に通知される。例えば、回転の割合は、式3で計算できる。

(現在の2つの指先の絶対的な角度 - ペアリング時点の2つの指先の絶対的な角度) ... (式3)

出力制御部127は例えば、ROTATEイベントの情報を用いて、表示アイテムを回転させることができる。

## 【0067】

## &lt;マルチタッチ操作の具体例&gt;

マルチタッチ操作を、距離画像センサ115を用いたタッチ認識で実現する場合には、操作に用いられる複数の指先がそれぞれタッチ状態であることが認識されている必要がある。マルチタッチ操作の認識においても、操作入力に用いられない指の指先がタッチ状態であると誤認識されると、誤動作の原因となる可能性がある。

## 【0068】

まず、本実施形態の絞り込みを行わずに全ての指先の高さを、一定のタッチ閾値200と比較する場合、マルチタッチ操作が入力される場合に起こり得た問題を、図7(A)と図6(A)を参照して説明する。図7(A)はUI部品102に対してマルチタッチ操作をするために、手103aの人差し指303と親指304でタッチした瞬間の様子である。拡大・縮小操作や回転操作を行う場合には人差し指と親指が用いられることが多いが、ユーザにとって、人差し指と親指だけを伸ばし、他の指が誤検出されないように曲げたり向きを変えたりする姿勢を取ることは負担を伴うことがある。従って、ユーザには、図7(A)のように、手及び指を自然と開いた姿勢でマルチタッチ操作を行える環境が期待される。

## 【0069】

図6(A)は図7(A)における指先高さを、z×平面にマッピングしたものである。図4及び図5で示したのと同じ要素については、同じ番号を付与し、説明を省略する。この場合、ユーザが操作入力に用いることを意図したのは人差し指303と親指304であるが、他の指も広げられた状態にあるため、人差し指303と親指304の動きに連動して動く。図6(A)の場合、人差し指303と親指304とは離れた小指300が連動して下がったことで、小指300の指先高さがタッチ閾値200を下回っている。従って、例えば、誤って小指300が親指304あるいは人差し指303とペアリングされてしまい、ユーザの意図と異なるマルチタッチ操作が認識されることが有り得る。

## 【0070】

本実施形態では、上述したように、タッチ閾値と指先高さの比較によるタッチ認識処理を行う前に、タッチ認識処理の対象とする指先を絞り込む。図6(B)に、図6(A)と同様の状況で、本実施形態の絞り込みを適用した結果を示す。図6(B)に示す手の状態では、複数の指先の中でも人差し指303の指先高さが最も小さい。その指先高さがH<sub>min</sub>となる。操作入力に用いられる人差し指303と親指304は、絞り込み閾値305とな

10

20

30

40

50

る高さ $H_{min} + \Delta H$ を下回っているため、タッチ認識の対象となる。そしてタッチ閾値200を下回っているので、タッチ状態であると認識される。一方で、操作入力に用いられる指ではない小指300の指先高さは、タッチ閾値200を下回っているものの、絞り込み閾値305である高さ $H_{min} + \Delta H$ を上回っているため、タッチ閾値200との比較が行われない。従って、タッチ状態であるとは認識されない。

#### 【0071】

次に指先の移動スピードに基づく指先の絞り込みの例を図6(C)を参照して説明する。図6(C)では図5(A)と同様に前フレームの指先位置を破線の円で、現フレームの指先位置を実線の円で示し、前フレームと現フレームの指先高さ位置を繋いだ矢印はZ軸方向のスピードの大きさを示している。全ての指先の中で最も指先スピードが大きい指先は人差し指303である。その指先スピードを $V_{max}$ とする。指先スピードが $V_{max} - \Delta V$ を上回っているのは人差し指303と親指304である。従って、本実施形によれば、人差し指303と親指304がタッチ認識処理の対象となり、それらはタッチ閾値300より高さが小さいため、タッチ状態であると認識される。一方、小指300の指先高さは、タッチ閾値200を下回っているものの、指先のスピードが $V_{max} - \Delta V$ を下回っているため、タッチ認識の対象とされない。従って、小指300はタッチ状態であると認識されない。

#### 【0072】

図7(B)は、本実施形態によって指先のペアリングが正しく行われたことにより、縮小操作が認識され、UI部品102が縮小された様子を表す。人差し指303と親指304の距離を小さくすることでPINCHイベントが発行され、アプリケーションはPINCHイベントの拡大・縮小割合を参照し、UI部品102の大きさを変化させている。図7(C)は、UI部品102に回転操作を行われた様子を表す。人差し指303と親指304を時計回りに回転させることでROTATEイベントが発行され、アプリケーションはROTATEイベントの回転の割合を参照し、UI部品102を時計回りに回転させている。このように、本実施形態では、一部の指だけを使ったマルチタッチ操作を行った場合でも、適切な指の情報を用いてマルチタッチ操作が認識されるので、ユーザが自然と手を広げた楽な姿勢で操作を行うことができる。

#### 【0073】

なお、本実施形態では、ユーザが一人で片手でのマルチタッチ操作を行うことを前提として説明して来た。従って、高さやスピードの比較は、同一の手に属する複数の指先間で行われた。ただし、マルチタッチ操作は、両手で行われる場合もあり得る。また、右手での操作中に、別途左手の誤タッチが検出されてしまうといった場合があり得る。そのような場合に対応するために、抽出された複数の手領域の中で同一ユーザの両手を組み合わせを特定し、1ユーザの両手に属する指先の中で高さやスピードを比較してもよい。こうすることで、指だけでなく手に関しても、ユーザが操作に用いる意図を持っているか否かを推定することで、操作の誤認識を低減することができる。また、複数ユーザの指先の高さやスピードを全て比較して絞り込みを行うことで、あるユーザが主体的に操作している場合に、他のユーザの指先による誤タッチが認識されてしまうことを防ぐことができる。また、上述した例ではマルチタッチ操作に用いられる指先は2つずつペアリングしたが、3本以上の指先が協働してマルチタッチ操作を行う場合にも、本実施形態は有効である。このように、意図した人差し指303と親指304のみがタッチ状態になると、この2つの指がペアリングされる。そして、人差し指303と親指304をタッチ状態のまま動かすことによって、マルチタッチ操作を行うことができる。

#### 【0074】

##### <平面以外へのタッチ操作>

ここまで、平面を対象としたタッチ操作を例に挙げて説明してきたが、本実施形態は、任意形状の物体が対象であっても同様の方法で誤認識を抑制したタッチ操作を行うことが可能である。一例として、平面上におかれた球500を対象にタッチ操作を行う場合の例を図13(A)に示す。球400を覆うように、一定のタッチ閾値401、リリース閾値402を設定する。そして、指先403の球400の表面までの距離404が、最小とな

10

20

30

40

50

る指先の位置情報を基準として絞り込みを行う。また、指先の移動スピードを取得する際には、指先 403 が球 400 の表面へ向かう方向のスピード 405 を比較すればよい。

#### 【0075】

また、仮想オブジェクトが対象であっても、同様の方法で誤認識を抑制したタッチ操作を行うことが可能である。図 13 (B) を用いて、3D プロジェクタで空中に投影された 3D オブジェクトである球 406 に対するタッチ操作の例を示す。この場合も、球 406 を覆うようにタッチ閾値 407 とリリース閾値 408 を設定する。そして、指先 409 の球 406 の表面までの距離 410 が最小距離となる乳先指先高さ 410 が最小となる指先の位置情報を基準として絞り込みを行う。また、指先の移動スピードを取得する際には、指先 409 が球 406 の表面へ向かう方向のスピード 411 を比較すればよい。

10

#### 【0076】

##### [変形例]

上述した第 1 の実施形態では、絞り込まれた指先がタッチ状態であるかを判定する条件は、予め設定された一定のタッチ閾値及びリリース閾値であった。それに対し変形例として、ユーザ毎に、あるいは指毎に、より適切な閾値を決定する例を説明する。なお、以下の変形例で説明するタッチ閾値の決定処理は、図 8 のフローチャートのステップ S105 の処理において行われる。変形例では、ステップ S105 以外の処理は、第 1 の実施形態に準じるため、詳細な説明は省略する。

#### 【0077】

##### <変形例 1：指同士の関係に基づく閾値の補正処理>

20

第 1 の実施形態、及びその変形例は、操作に利用される指に連動して動いてしまう他の指の影響により生じるタッチ操作の誤認識を低減することを目的としている。ここで、操作に利用される指の種類によっては、それに連動して動いてしまう傾向が強い指を予め想定することが可能である。例えば、ユーザやアプリケーションの都合で、小指でタッチ操作が行われる場合、小指と隣り合う薬指は、小指の動きに連動して動いてしまいやすい。またその際、入力操作に用いられる小指と、高さやスピードが近いものになることが統計的に解っている。ここで変形例 1 では、このような指同士の関係に基づいて閾値を決定する。

#### 【0078】

図 5 (B) は小指 300 でタッチした場合の様子である。このとき薬指 301 は、指先高さおよび指先スピードが小指 300 とほぼ等しいため、第 1 の実施形態の絞り込みによって除外することが難しい場合がある。そこで変形例 1 では、小指 300 と連動して動く傾向が強い薬指 301、及び中指 302 に関しては、小指 300 がタッチ操作に用いられる可能性が最も高い指だとみなされた場合に用いるタッチ閾値及びリリース閾値を先に補正しておく。なお、第 1 の実施形態の場合、タッチ操作に用いられる可能性が最も高い指だとみなされるのは、最もタッチ対象面からの距離が近く、また移動スピードが速い指である。

30

#### 【0079】

図 5 (C) は、小指 300 が最もタッチ対象面からの距離が近く、また移動スピードが速い指であったことに応じて、薬指 301、及び中指 302 のタッチが認識される条件が厳しくなった様子を表す。小指 300 と連動して動く薬指 301 のタッチ閾値 306 は、通常のタッチ閾値 200 よりも小さい値に決定されている。同様に中指 302 のタッチ閾値が、閾値 200 から閾値 307 に補正される。最終的に、指先高さおよび指先スピードを比較することによって、小指 300 と薬指 301 に対してタッチ判定が行われるが、薬指 301 の指先高さは補正されたタッチ閾値 306 を下回っていないためタッチ状態であると認識されない。意図したタッチである小指 300 だけがタッチ状態であると認識される。

40

#### 【0080】

図 12 (A) は上述した、指同士の関係に基づく閾値の決定処理の流れを表すフローチャートである。この処理はステップ S105 の中に実行される。ステップ S500 におい

50

て、タッチ認識部 126 は、指先高さが最小の指先を特定し、その指先と連動して動く指先の閾値を決定する。ステップ S301において、タッチ認識部 126 は、指先速度が最大の指先を特定し、その指先と連動して動く指先の閾値を決定する。ステップ S500 及びステップ S501 では例えば、予め設定されているタッチ閾値に、所定の補正量を加えるあるいは減じることで、補正された閾値を算出し、対象としている指の閾値として決定する。ただしステップ S300 で決定済みの指先は決定しない。一例として、ステップ S300 およびステップ S301 で、連動して動く指先および補正量を取得するために、図 12 (B) に示すルックアップテーブルを用いる。

#### 【0081】

なお変形例 1 では、連動して動く指先および補正量を取得するために図 12 (B) に示すルックアップテーブルを用いるが、連動して動く指先および補正量を決めるために他の方法を用いることも可能である。例えば、統計情報に基づいて確率的に決めることができる。なお、変形例 1 では指の種類の情報を使って、連動して動きやすい関係にある指先を特定しタッチ閾値を決定しているが、そのような情報がない場合でも、連動して動く指先を特定することは可能である。例えば、指先高さが低いあるいは指先速度が大きい指先に隣接する指先を、連動して動く指先であると判定し、その指先の閾値を決定することができる。

10

#### 【0082】

<変形例 2：ユーザのニュートラルポジションに合わせた指毎の閾値補正>

第 1 の実施形態では、どのようなユーザについても一定のタッチ閾値、及びリリース閾値を使った。しかしながら、ユーザによって、ニュートラルポジション（自然な手の状態）での各指の高さは異なる。そこで、ユーザの手の状態がニュートラルポジションにあるときの各指の高さに基づいて閾値を補正する場合を説明する。

20

#### 【0083】

図 14 (B) はニュートラルポジションの指先高さによって閾値を変動させた様子を示している。変形例 2 では、条件決定部 124 は、手がニュートラルポジションにあることが確認できた状態で得られた、各指先の高さ情報を使って閾値を補正する。例えば、手のシルエットを投影し、ユーザにそのシルエット上に手を静止させてもらうガイダンス表示を使って、ユーザの手をニュートラルポジションに誘導してから図 8 のフローチャートの処理を開始する。この場合、シルエット上に指先が 5 つ検出され、一定時間経過した場合にニュートラルポジションになったことを認識することができる。

30

#### 【0084】

そして、条件決定部 124 は、ニュートラルポジションにある手の各指先の高さを予め設定した高さ閾値 と比較する。ここで用いる高さ閾値 は、タッチ閾値及びリリース閾値とは異なるものであり、これ以上高い位置にある指は、タッチ操作に関わる可能性が低いとみなされ、タッチ閾値の補正が必要ではないと判定するための基準の高さである。図 14 (B) において、薬指 301、中指 302、人差し指 303 は指先高さが一定の高さより高いため、タッチ閾値 200 とリリース閾値 201 はデフォルトのまま変化させない。しかし小指 300 はニュートラルポジションにおける指先高さが低いため、その高さに合わせてタッチ閾値を 510、リリース閾値を 511 と再設定する。同様に、親指 304 もタッチ閾値を 512、リリース閾値を 513 と再設定する。変形例 2 では、このように、ユーザに合わせて閾値を設定することで誤認識が少ない自然なタッチ操作が可能になる。

40

#### 【0085】

ただし、所定の高さ による区別を行わず、全ての指について、ニュートラルポジションでの指の高さ応じて閾値を変えてもよい。図 14 (A) は指の種類に応じて閾値を設定した例を示している。小指 300 のタッチ閾値は 500、リリース閾値は 501 である。同様に、薬指 301 のタッチ閾値は 502、リリース閾値は 503 である。中指 302 のタッチ閾値は 504、リリース閾値は 505 である。人差し指 303 のタッチ閾値は 506、リリース閾値は 507 である。親指 304 のタッチ閾値は 508、リリース閾値は 5

50

09となる。このように詳細に閾値を合わせることで、どの指先でも直観的に同程度の認識率と操作感を実現することができる。

#### 【0086】

＜変形例3：アプリケーションに応じた指毎の閾値補正＞

また別の変形例3として、ニュートラルポジションでの高さにはよらず、ステップS103で特定した種類に応じて、所定の指のタッチ閾値を調整してもよい。例えば、アプリケーションによっては、所定の種類の指先を認識しやすくすることで、自然な操作感を実現できる場合がある。人差し指と親指によるマルチタッチ操作を頻繁に行うアプリケーションにおいて、人差し指と親指を認識しやすくし、それ以外の指を認識しにくくする場合を説明する。図14(C)は、小指300、薬指301、中指302に関して、タッチを検出する条件が厳しくなるように、閾値を設定した様子を表す。まず、小指300、薬指301、中指302のタッチ閾値514およびリリース閾値515を低く設定する。また、人差し指303および親指304のタッチ閾値516、リリース閾値517を高めに設定することでタッチを認識しやすくしている。

10

#### 【0087】

また、マルチタッチ操作を行う場合には、操作する以外の指が折り曲げられていると、5つ全ての指先が検出できず、指先の種類の認識結果が得られない場合がある。このような場合、指が検出できていなくても、ステップS103での指の特定結果を利用して、各指に最適な閾値を設定することができる。例えば、右手であれば、侵入位置から見て左(図14(D)においては右)から1番目の指先と2番目の指先がそれぞれ親指と人差し指であると思われる所以、それらの指先を認識しやすくできる。図14(D)は右手の4つの指先518、519、520、521が検出されているが、種類は分からず状態を示している。ここで指先520と521は人差し指と親指であると思われる所以、高めのタッチ閾値516、リリース閾値517が設定されており、指先518と519には低めのタッチ閾値514およびリリース閾値515が設定されている。同様に、左手であれば、侵入位置から見て右から1番目の指先と2番目の指先がそれぞれ親指と人差し指であると思われる所以、それらの指先を認識しやすくできる。

20

#### 【0088】

以上説明した変形例1～3によって、タッチ閾値を決定する場合、決定されたタッチ閾値に合わせてリリース閾値を決定することで、タッチ操作認識処理の全体の精度をより向上させることができる。変形例においても、第1の実施形態で説明したように、決定されたタッチ閾値及びリリース閾値を使ってタッチ状態が否かを判定する指先を絞り込むことで、操作に利用される指に連動して動いてしまうにより生じるタッチ操作の誤認識を低減することができる。

30

#### 【0089】

(その他の実施形態)

本発明は、上述の実施形態の1以上の機能を実現するプログラムを、ネットワーク又は記憶媒体を介してシステム又は装置に供給し、そのシステム又は装置のコンピュータにおける1つ以上のプロセッサーがプログラムを読み出し実行する処理でも実現可能である。また、1以上の機能を実現する回路(例えば、ASIC)によっても実現可能である。

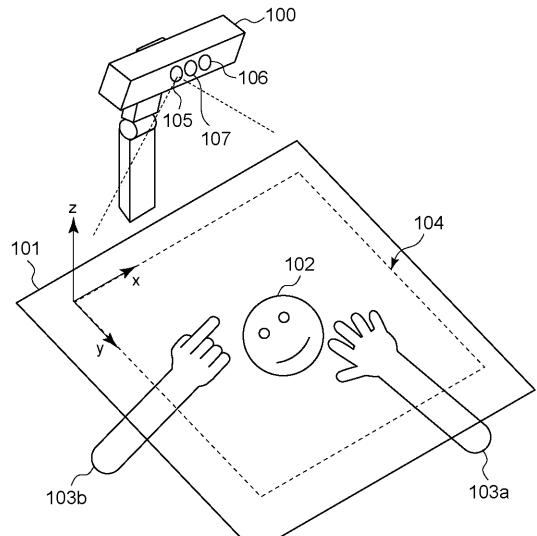
40

#### 【符号の説明】

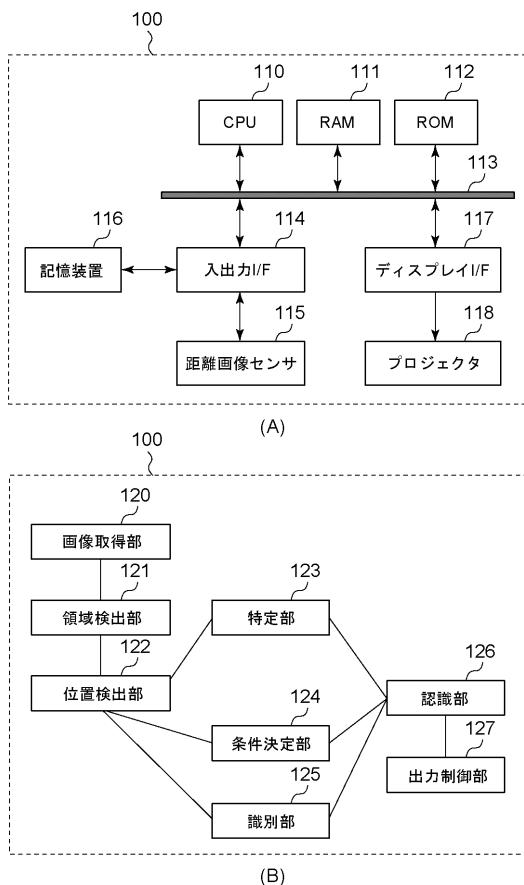
#### 【0090】

- 120 画像取得部
- 121 領域検出部
- 122 位置検出部 123 特定部
- 124 条件決定部
- 125 識別部
- 126 認識部
- 127 出力制御部

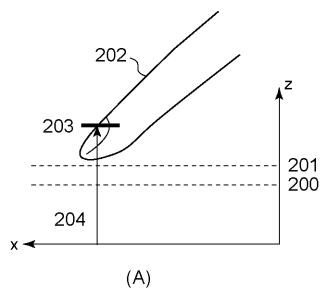
【図1】



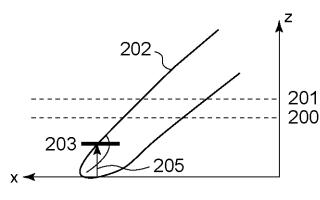
【図2】



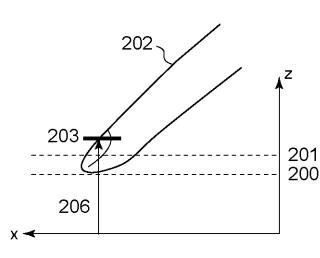
【図3】



(A)

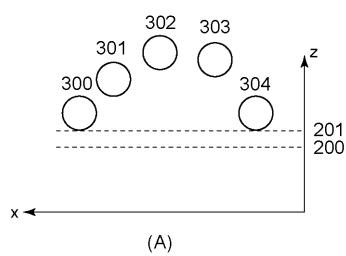


(B)

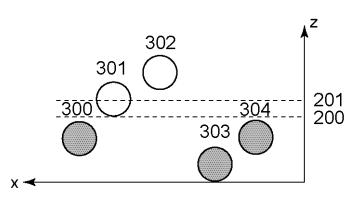


(C)

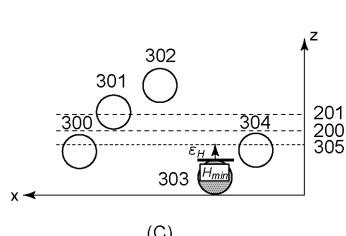
【図4】



(A)

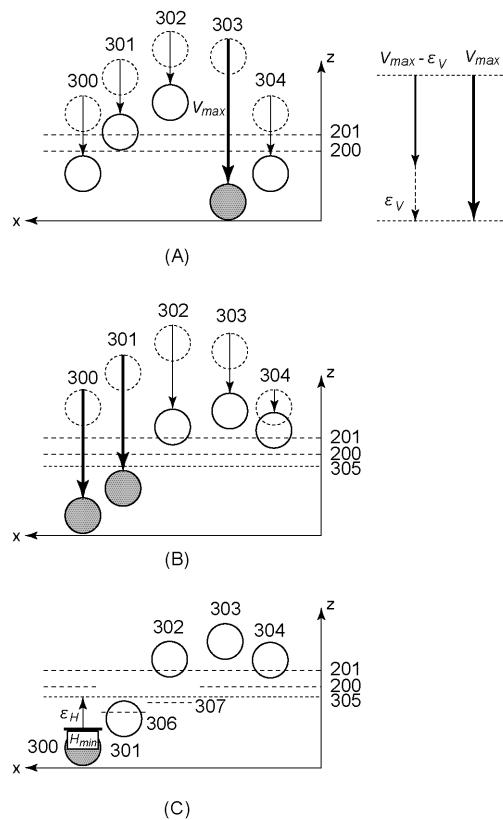


(B)

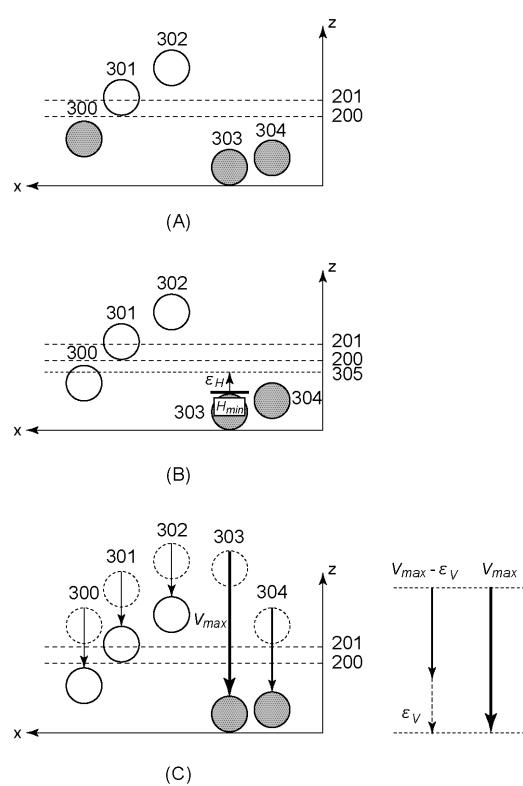


(C)

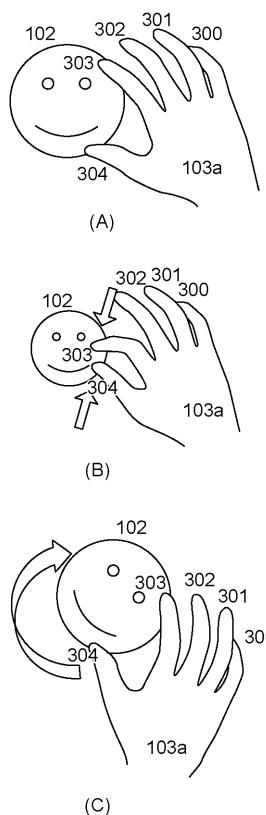
【図5】



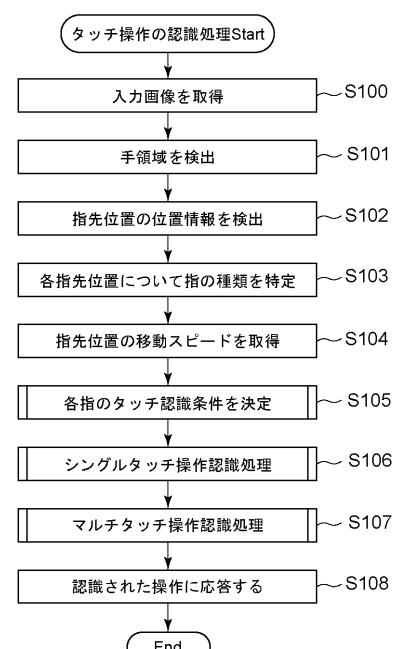
【図6】



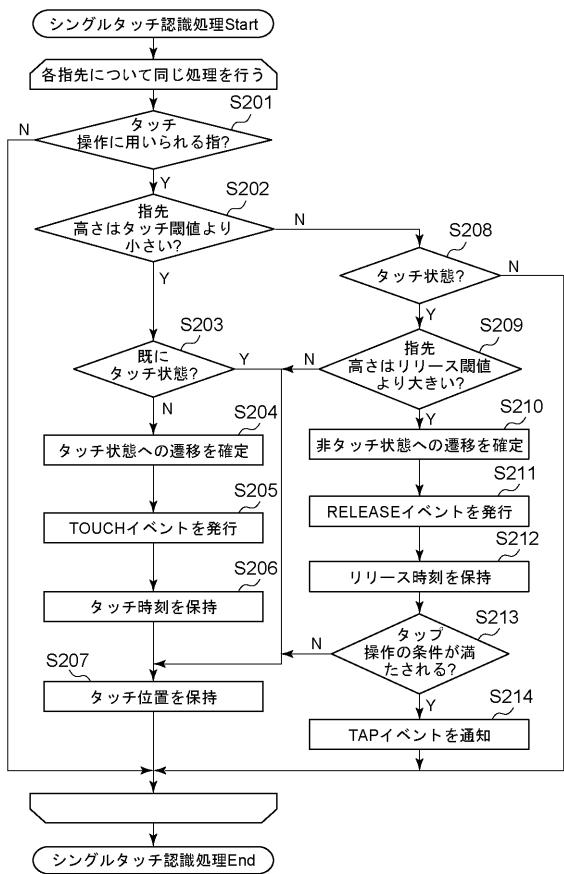
【図7】



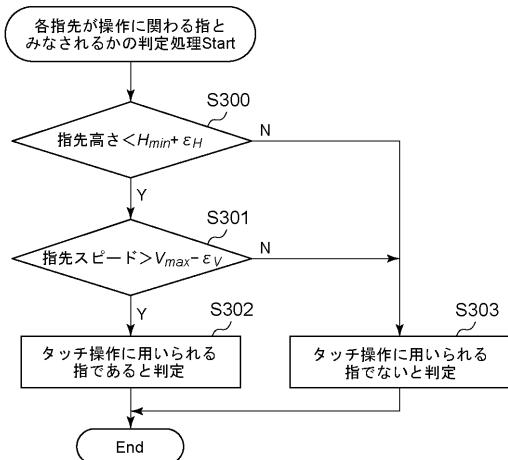
【図8】



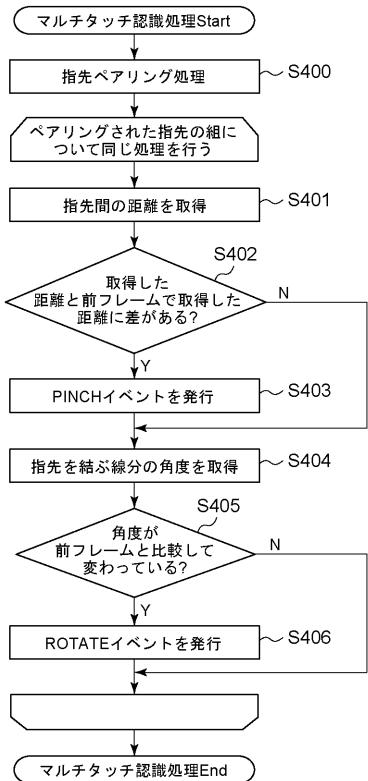
【図9】



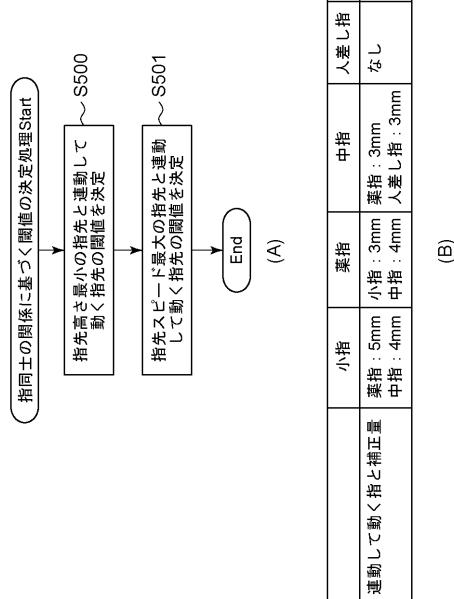
【図10】



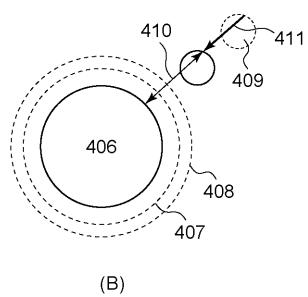
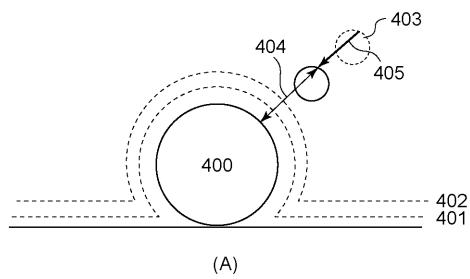
【図11】



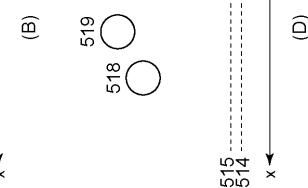
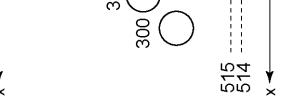
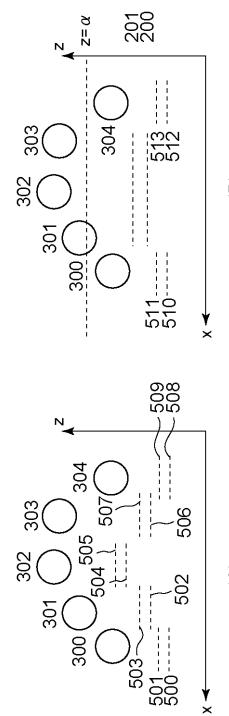
【図12】



【図13】



【図14】



---

フロントページの続き

(56)参考文献 特表2015-519624(JP,A)  
特開2012-108857(JP,A)  
米国特許出願公開第2014/0029789(US,A1)  
特開平11-305895(JP,A)  
特開2015-102983(JP,A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 3 / 041  
G 06 F 3 / 042