

(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2005-352738

(P2005-352738A)

(43) 公開日 平成17年12月22日(2005.12.22)

(51) Int. Cl.⁷

G06F 3/033

F I

G06F 3/033 380D

テーマコード(参考)

5B087

審査請求 未請求 請求項の数 7 O L (全 11 頁)

(21) 出願番号 特願2004-172321 (P2004-172321)

(22) 出願日 平成16年6月10日(2004.6.10)

特許法第30条第1項適用申請有り 平成16年3月8日 社団法人電子情報通信学会発行の「E I C 電子情報通信学会 2004年総合大会講演論文集」に発表

(71) 出願人 504224201

有限会社アイ・テクノロジー

大阪府吹田市山田東4-8-3-104

(74) 代理人 100123504

弁理士 小倉 啓七

(74) 代理人 100127166

弁理士 本間 政憲

(72) 発明者 北村 喜文

吹田市山田東4-1-1-606

Fターム(参考) 5B087 AA01 BB01 DD03

(54) 【発明の名称】 ターゲット位置予測を用いたポインティング操作システム

(57) 【要約】

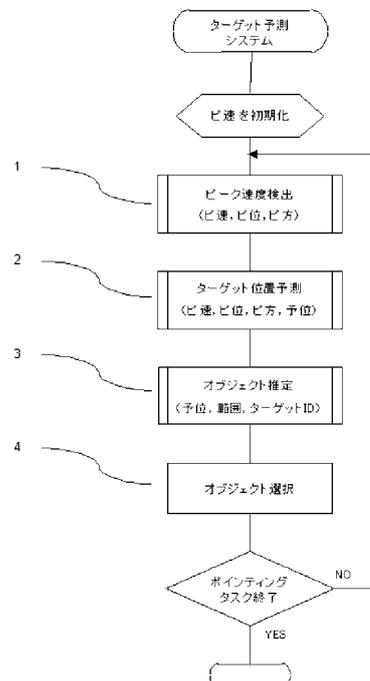
【課題】

大画面化、高解像度の画面上および仮想空間内のGUI環境下において、マウス、3Dトラッカー等のポインティング操作の時間を短縮できるインターフェイスを提供する。

【解決手段】

ポインティング操作において、ポインタ移動中のピーク速度とその時点でのターゲットまでの距離に相関があり、ピーク速度からターゲットまでの距離を求め、ポインタ位置、方向とからターゲット位置を予測することで、ポインティング操作時に、ユーザがポインタを移動させたいオブジェクト位置を、システム側が操作完了前に予測できることにより、その場所に存在するアイコンが示すアプリケーションを起動し始める等して、作業時間を短縮できるシステムを構築する。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

情報端末装置の画面上でマウスなどのポインティングデバイスを利用してマウスカーソルなどのポインタを移動させる場合または仮想現実空間内で3次元トラッカーなどを操作してポインタを移動させる場合において、

前記ポインタ位置もしくはポインタ速度の情報を取り込み、

2) 前記ポインタ速度が増加から減少に転じる点としてポインタ速度の極大値(ピーク速度)を検出し、

3) 前記ピーク速度からターゲットまでの予測距離を1次相関関数によって算出し、

4) 前記ターゲットまでの予測距離とピーク速度時のポインタ位置とポインタ移動方向からターゲット予測位置を算出し、

前記ターゲット予測位置から一定範囲に存在するオブジェクト(アイコンなど)のうち1のオブジェクトを選択できることを特徴とするポインティング操作システム。

【請求項 2】

前記ポインタ速度の極大値(ピーク速度)が検出される度にターゲット位置を予測するが、前記ポインタ速度の極大値(ピーク速度)を検出する手段において、複数の極大値が検出された場合に、2回目以降に検出した極大値がそれ以前の極大値より小さかった場合には、その値をピーク速度として採用しないことを特徴とする前記請求項1記載のポインティング操作システム。

【請求項 3】

前記ピーク速度の時のポインタの速度ベクトルを画面上若しくは空間内のX軸方向とY軸方向とZ軸方向の成分にベクトル分解し、各方向毎にターゲットまでの距離を予測し、ターゲットの予測位置を算出することを特徴とする前記請求項1および2記載のポインティング操作システム。

【請求項 4】

前記ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトに応じたアプリケーションプログラムを自動的に起動することを特徴とする前記請求項1乃至3記載のポインティング操作システム。

【請求項 5】

前記ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトの位置にポインタを自動的に移動させることを特徴とする前記請求項1乃至3記載のポインティング操作システム。

【請求項 6】

前記ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトのアイコンの形態を変化させることを特徴とする前記請求項1乃至3記載のポインティング操作システム。

【請求項 7】

前記ピーク速度からターゲットまでの距離を予測する場合において、個々のユーザ毎にピーク速度からターゲットまでの距離を求める換算式の係数を、予め数回の試行によって決定するキャリブレーション機能を有することを特徴とする前記請求項1記載のポインティング操作システム。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は、情報端末装置の画面上または仮想現実空間内でのポインティング操作予測技術に関し、特にマウス、三次元トラッカー等のポインティングデバイスの操作に関するものである。

【背景技術】**【0002】**

10

20

30

40

50

近年、情報処理装置画面は大画面化、高解像度のディスプレイが採用されている。このような GUI (Graphical User Interface) 環境下では、ポインティング操作に必要な時間が増加する傾向にあり、今後この傾向は益々顕著になると推定される。このためマウス等のポインティング操作の時間を短縮できるインターフェイスが必要とされる。

【0003】

従来、現在のポインタの位置を元にその近傍領域を検索領域とし、検索領域範囲内に操作対象となるオブジェクトが存在するかをチェックし、存在する場合に自動的にそのオブジェクトの位置にポインタを移動させるものがあった(特許文献1)。

【0004】

また、画面上のポインタ移動方向を示す直線上に操作対象となるオブジェクトが存在するかをチェックし、存在する場合に自動的にそのオブジェクトの位置にポインタを移動させるものがあった(特許文献2)。

【0005】

また、画面上でポインタを移動させる場合に、ポインタの位置、移動方向、および移動速度の情報に応じて、ポインタの移動先の検索範囲を想定し、検索範囲内から現在のポインタ位置に最も近い位置にあるオブジェクトを移動先オブジェクトと予測し、ポインタを自動的に移動させるものがあった(特許文献3)。

【0006】

さらに、あらかじめマウスの速度プロファイル関数を準備し、マウスの動き出し速度の傾きから最も近似する速度プロファイル関数を抽出して、それに基づき移動先オブジェクトと予測し、ポインタを自動的に移動させるものがあった(特許文献4)。

【0007】

【特許文献1】特開平8-76956号公報

【特許文献2】特開平6-83540号公報

【特許文献3】特開平10-222307号公報

【特許文献4】特開平9-244808号公報

【発明の開示】

【発明が解決しようとする課題】

【0008】

第1の課題は、上記特許文献1に記載されている技術において、現在のポインタ位置の近傍を検索領域とする方法では、ユーザがポインタを移動させたいオブジェクトと現在のポインタ位置との距離が大きい場合、多くの回数のポインティングデバイス操作が必要となり、操作性が好ましくないことである。

【0009】

第2の課題は、上記特許文献2に記載されている技術において、現在のポインタ移動方向の延長線上にオブジェクトを検索する方法では、ユーザがポインタを移動させたいオブジェクトと現在のポインタ位置との距離が小さい場合、ポインタを移動させたいオブジェクトが現在のポインタ位置の近くにあるにもかかわらずポインタを移動させる方向によっては全く目的としていない離れたオブジェクトへポインタが移動してしまう可能性があり、操作性が好ましくないことである。

【0010】

第3の課題は、上記特許文献3に記載されている技術において、ポインタの移動速度の情報に応じてポインタの移動先の検索範囲を想定しオブジェクトを予測する方法では、移動速度に応じて検索範囲を広げ設定するため、ユーザがポインタを移動させたいオブジェクトと現在のポインタ位置との間に別のオブジェクトが存在すれば、オブジェクトの選択ができず、操作性が好ましくないことである。

また、特許文献3に開示されている方法では、移動速度が0になったとき、すなわち、速度プロファイルが完結したときに予測動作が行われるため、予測するためにはマウスなどのポインティングデバイス操作を必ず停止させる必要があった。

さらに、特許文献3に開示されている方法では、ポインタ移動方向がポインタの停止直

10

20

30

40

50

前の移動方向の延長線上になるため、素早くポインティングデバイス进行操作している場合には、停止直前の移動方向はノイズが多くなり、ターゲットオブジェクトの方向と一致しない場合が多い。

【0011】

第4の課題は、上記特許文献4に記載されている技術において、速度プロファイルを使っているが、今の動きがどの速度プロファイルと似ているかを動き始めの移動速度の立ち上がりにより判断しているため、ブレや誤差が大きくなっていることである。最初の動きだしではノイジであり、特に人のうごきはジャギーに動くため、最初の動きで判断する方法は好ましくない。

また、あらかじめ状況に応じた速度プロファイルの関数群を用意しておく必要があった。

【課題を解決するための手段】

【0012】

本発明者はヒューマンインターフェイスについて鋭意に研究を行った結果、ポインティング操作時に、ユーザがポインタを移動させたいオブジェクト位置を、システム側が操作完了前に予測できることにより、その場所に存在するアイコンが示すアプリケーションを起動し始める等して、作業時間を短縮し、また、ユーザの作業負担を軽減できる可能性があり、非常に有効であることを知見した。

【0013】

また、本発明者は、ポインティング操作において、ポインタ移動中のピーク速度とその時点でのターゲットまでの距離に相関があることに着目した。すなわち、下記式により、ポインタ移動中のピーク速度とその時点でのターゲットまでの距離を表すことができ、下記式を利用して、ピーク速度からターゲット位置を予測できることを知見し、本発明を完成するに至った。

【0014】

(数1)

ターゲットまでの予測距離 = $a \times \text{ピーク速度} + b$

【0015】

ここで、 a 、 b はキャリブレーションなどの試行により求まるパラメータ係数であり、ポインティング操作を行う個々の操作者によって異なる値をとる。

【0016】

本発明に係るポインティング操作システムの第1の観点によれば、「情報端末装置の画面上でマウスなどのポインティングデバイス进行操作してマウスカーソルなどのポインタを移動させる場合または仮想現実空間内で3次元トラッカーなどを操作してポインタを移動させる場合において、前記ポインタ位置もしくはポインタ速度の情報を取り込み、前記ポインタ速度が増加から減少に変化する点としてポインタ速度の極大値(ピーク速度)を検出し、前記ピーク速度からターゲットまでの予測距離を1次相関関数によって算出し、前記ターゲットまでの予測距離とピーク速度時のポインタ位置とポインタ移動方向からターゲット予測位置を算出し、前記ターゲット予測位置から一定範囲に存在するオブジェクト(アイコンなど)のうち1のオブジェクトを選択できる」機能が提供される。

【0017】

ここで、1次相関関数とは上述の数1で示される数式である。ピーク速度を三次元ベクトルで示すことにより、二次元の情報端末画面上のみならず、三次元の仮想空間内でのポインティングデバイスに適用できる。

【0018】

また、ターゲット予測位置から一定範囲に存在するオブジェクトとは、ターゲット予測位置から一定距離範囲内にあるアイコンなどのオブジェクトをいう。この一定距離を決定するパラメータは分解能に密接に影響するため可変パラメータとしている。

【0019】

また、一定範囲に存在するオブジェクト(アイコンなど)のうち1のオブジェクトを選

択する方法は、例えば、ターゲット予測位置から最も近いオブジェクトを選択したり、また、使用頻度の高いものを選択したり、その時の状況やコンテキストに合ったオブジェクトを選択したりするなどの方法が考えられる。

【0020】

本発明に係るポインティング操作システムの第2の観点によれば、前記第1の観点において、「ポインタ速度の極大値（ピーク速度）が検出される度にターゲット位置を予測するが、ポインタ速度の極大値（ピーク速度）を検出する手段において、複数の極大値が検出された場合に、2回目以降に検出した極大値がそれ以前の極大値より小さかった場合には、その値をピーク速度として採用しない」機能が提供される。

【0021】

操作途中の最加速や手の震えなどにより複数の極大値が検出されると考えられるためである。

【0022】

本発明に係るポインティング操作システムの第3の観点によれば、前記第1および第2の観点において、「ピーク速度の時のポインタの速度ベクトルを画面上若しくは空間内のX軸方向とY軸方向とZ軸方向の成分にベクトル分解し、各方向毎にターゲットまでの距離を予測し、ターゲットの予測位置を算出する」機能が提供される。

【0023】

素早くポインティングデバイス进行操作している場合には、停止直前の移動方向はノイズが多くなり、ターゲットオブジェクトの方向と一致しない場合が多いことを考慮して、軸方向にポインタ速度成分を分解し、ピーク速度時におけるポインタ軌道の接線方向を移動方向と認識することにより、ターゲット予測位置のブレを最小限に抑えることとするものである。

【0024】

本発明に係るポインティング操作システムの第4の観点によれば、前記第1から第3の観点において、「ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトに応じたアプリケーションプログラムを自動的に起動する」機能が提供される。

【0025】

本発明に係るポインティング操作システムの第5の観点によれば、前記第1から第3の観点において、「ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトの位置にポインタを自動的に移動させる」機能が提供される。

【0026】

本発明に係るポインティング操作システムの第6の観点によれば、前記第1から第3の観点において、「ターゲットの予測位置から一定範囲に存在するオブジェクトのうち1のオブジェクトを選択できた場合に、選択されたオブジェクトのアイコンの形態を変化させる」機能が提供される。

【0027】

本発明に係るポインティング操作システムの第7の観点によれば、前記第1の観点において、「ピーク速度からターゲットまでの距離を予測する際に、個々のユーザ毎にピーク速度からターゲットまでの距離を求める換算式の係数を、予め数回の試行によって決定する」キャリブレーション機能が提供される。

【発明の効果】

【0028】

本発明に係るポインティング操作システムによれば、ユーザがポインタを移動させたいオブジェクトと現在のポインタ位置との距離が大きい場合においても、少ない回数 of ポインティングデバイス操作で十分であり、操作性が向上する。

【0029】

また、本発明に係るポインティング操作システムによれば、ポインタを移動させたいオ

10

20

30

40

50

プロジェクトが現在のポインタ位置の近くにある場合、通常、ポインティングデバイスを緩慢に操作することとなり、ピーク速度が小さくなり、ターゲット予測位置は近傍となるため、全く目的としていない離れたオブジェクトへポインタが移動してしまう可能性はなく、操作性が向上する。

【0030】

また、本発明に係るポインティング操作システムによれば、ユーザがポインタを移動させたいオブジェクトと現在のポインタ位置との間に別のオブジェクトが存在しても、オブジェクトの選択ができないということはない。また、速度プロファイルが完結する前に予測が行えるため、より迅速なポインティング操作に対応できる。

【0031】

さらに、ポインティング操作のピーク時速度から直接ポインティング予想位置を算出できるため、あらかじめ速度プロファイルの関数群を用意しておく必要はない。

【発明を実施するための最良の形態】

【0032】

以下に本発明の実施形態を説明するが、本発明は特にこれらに限定されるものではない。

【実施例1】

【0033】

以下、情報端末装置の画面上のターゲット位置予測を用いたポインティング操作に関して、処理フロー図を示しながら詳細に説明する。通常、情報端末装置のGUI環境下では、代表的にはマウスで、画面上のポインタ操作を行っている。操作者はポインタの行き先ターゲットを確認しながら、そのターゲット位置にポインタを近づけるように操作する。

【0034】

本発明者は、ポインティング操作において、ポインタ移動中のピーク速度とその時点でのターゲットまでの距離に相関があることを知見し、ピーク速度からターゲット位置を予測することで、ポインティング操作時間やターゲットのオブジェクトに対するアクション時間（例えば、アプリケーション・プログラムの起動時間）を短縮できるポインティング操作システムを完成した。

【0035】

本実施例に係るポインティング操作システムは、2つのプロセスから構成されている。1つは時々刻々と変化するポインタの変化を検知するプロセスで、具体的には、10msで周期的にマウスの位置座標を読み取るプログラムである。また、もう1つは、周期的にターゲット位置予測アルゴリズム処理を行うプロセスで、具体的には、マウスの位置座標を読み取るプログラムより遅い周期、例えば、100msでターゲット位置予測アルゴリズム処理を行うプログラムである。

【0036】

図1にターゲット位置予測のアルゴリズムを示す。ターゲット位置予測アルゴリズムは、時々刻々と変化するマウスポインタの位置座標から、ポインタのピーク速度を算出して、そのピーク速度からターゲットまでの距離を相関関数により算出し、ピーク速度を検出したときのポインタの座標および進行ベクトルから、ターゲット位置を予測している。

【0037】

ターゲット位置予測のアルゴリズムでは、初期化処理としてピーク速度を初期化して、図1の処理1～4をポインティングタスク（ターゲット位置予測アルゴリズム処理を行うプログラム）が終了するまで周期的に繰り返す。ポインタのピーク速度検出処理1は、ピーク速度を検出し、ピーク速度、ピーク位置、ピーク方向を求めている。次に、ターゲット位置予測処理2は、求めたピーク速度、ピーク位置、ピーク方向からターゲット位置を予測している。そして、オブジェクト推定処理3は、ターゲット予測位置から一定範囲内の距離にあるオブジェクトの中でターゲット予測位置に最も近いオブジェクトをターゲットと推定している。また、オブジェクト選択処理4は、選択されたオブジェクトに対して、例えば、起動命令を実行する処理である。また、起動と同時に、若しくは起動は行わず

10

20

30

40

50

に、選択されたオブジェクトのアイコンの形態を変化させることでもよい。

【0038】

図2は、ポインティング操作システムのピーク速度検出処理のフローチャートを示している。以下、ピーク速度検出処理について説明する。ステップS201は、用いられる変数を初期化している。ステップS202は、マウス等のポインティングデバイスからのシグナルを検知することにより、ポインタが移動したか否かを判定している。ステップS203は、サブルーチン関数である現在位置()、現在時刻()を用いて、ポインタの位置と時刻を求め、それぞれ位置B、時刻Bとしている。これら、現在位置()、現在時刻()は、処理をおこなうコンピュータのOS(オペレーティングシステム)等により提供されるものである。ステップS204は、位置Aが初期値であれば、ステップS208により、位置B、時刻Bを位置A、時刻Aに代入する。そして、ステップS205で、位置Aと位置Bの2点間の座標から、位置Aを始点とし位置Bを終点とする局所方向を算出している。また、前回までのポインタの方向を平均方向とし、今回算出した局所方向とのベクトル和を算出して、それで平均方向の値を更新する。ステップS206では、位置A、位置B、時刻A、時刻Bからポインタの速度を算出し、それを速度Bに代入している。ステップS207では、この速度Bを、これまでに蓄えられているピーク速度の値(ピ速)と比較し、速度Bがピーク速度以上であれば、S209で速度B、位置B、平均方向をそれぞれ、ピーク速度(ピ速)、ピーク速度位置(ピ位)、ピーク速度方向(ピ方)に代入する。速度Bがこれまでに蓄えられているピーク速度の値(ピ速)よりも小さければ、このピーク速度検出処理を終了する。

10

20

【0039】

図3は、ターゲット位置を予測する処理のフローチャートを示している。ステップS301は、ピーク速度の大きさとその方向から、ピーク速度のX軸方向、Y軸方向、Z軸方向の各成分を算出している。ここで、コンピュータ情報端末の画面の場合は、二次元座標となるので、水平成分(X軸方向)と垂直成分(Y軸方向)を求めることになる。次に、ピーク速度とターゲットまでの距離の相関式を用いて、速度からターゲットまでの予測距離を求める。ここで、マウスなどのポインティング操作を人が行う場合に、画面上で垂直方向にマウスを操作する場合と、水平方向にマウスを操作する場合とでは、相関式(ターゲットまでの距離 = a × ピーク速度 + b)のパラメータ係数a、bの値が異なってくる。このため、相関式を用いて予測距離を求める場合に、ピーク速度のX軸方向成分とY軸方向成分とZ軸方向成分の速度に分けて、各々異なるパラメータ係数a、bの相関式を用いることとしている。図3のステップS302は、ピーク速度のX軸方向成分を数1に代入して、X軸方向のターゲットまでの予測距離を算出している。また、同様に、ピーク速度のY軸方向成分を数1に代入して、Y軸方向のターゲットまでの予測距離を算出し、ピーク速度のZ軸方向成分を数1に代入して、Z軸方向のターゲットまでの予測距離を算出している。そして、ステップS303で、ピーク速度を検出した時のポインタ座標と、XYZ軸方向のターゲットまでの予測距離からターゲット予測座標を算出するのである。

30

【0040】

図4は、オブジェクト推定の処理のフローチャートを示している。まず、ステップS401で2つの変数(最短距離、ターゲット)を初期化している。ここで、最短距離は、予測位置から範囲内にある最も近いオブジェクトまでの距離を求めるための変数であり、またターゲットは、ターゲットと予測したオブジェクトのIDを求めるための変数である。ステップS403は、ターゲット予測位置座標と各オブジェクトの位置座標とから、ターゲット予測位置からオブジェクトまでの距離を算出している。ステップS404は、ターゲット予測位置からオブジェクトまでの距離が最短かどうかを判定するものである。ステップS405では、オブジェクトまでの距離が最短であった場合に、2つの変数である最短距離とターゲットの値を更新している。ここで、ステップS403からステップS405までは、ターゲット予測位置座標から一定距離範囲内に存在する全てのオブジェクトに対して行うべく繰り返し処理としている。

40

【0041】

50

次に、相関式（ターゲットまでの距離 = $a \times$ ピーク速度 + b ）のパラメータ係数 a , b の値を求めるキャリブレーションについて説明する。図 5 は、キャリブレーションの処理のフローチャートを示している。パラメータ係数の値は、個人によっても操作環境によっても異なるものである。あらかじめターゲットを決めておき、そのターゲットを選択するつもりで操作者にマウスを操作してもらうことを数回繰り返して、各ピーク速度、各ピーク位置、各ピーク方向から回帰分析を行い、ピーク速度からターゲットまでの距離を予測する式を生成するものである。パラメータ係数 a , b の決定に関しては、最小自乗法などを用いる。

【 0 0 4 2 】

下記の表 1、表 2 には、図 1 ~ 図 5 の処理フローチャートの中で用いられている変数、サブルーチン関数をまとめている。

10

【 0 0 4 3 】

【表 1】

変数の説明	
位置A, 位置B	ポインタの座標
局所方向	ポインタの進行方向ベクトル
平均方向	ポインタの平均進行方向ベクトル
速度B	ポインタの速度
時刻A, 時刻B	システムの時刻
ピ速	ピーク速度
ピ位	ピーク速度検出時のポインタの座標
ピ方	ピーク速度検出時のポインタの平均進行方向ベクトル
ピX速	ピーク速度のX成分
ピY速	ピーク速度のY成分
ピZ速	ピーク速度のZ成分
X距	ピーク速度のX成分から予測したターゲットまでのX方向の距離
Y距	ピーク速度のY成分から予測したターゲットまでのY方向の距離
Z距	ピーク速度のZ成分から予測したターゲットまでのZ方向の距離
予位	ターゲットの予測位置
範囲	オブジェクトをターゲットと予想するための予測位置からの一定距離
ターゲット	ターゲットと予測したオブジェクトのID
最短距離	予測位置から範囲内にある最も近いオブジェクトまでの距離
オブジェクト距離	予測位置からオブジェクトまでの距離

20

30

40

【 0 0 4 4 】

【表 2】

サブルーチンの説明	
ピーク速度検出(ピ速, ピ位, ピ方)	ピーク速度を検出し, ピ速, ピ位, ピ方に値を入れる。
位置予測(ピ速, ピ位, ピ方, 予位)	ピ速, ピ位, ピ方からターゲット位置を予測し, 予位に入れる。
ターゲット予想(予位, 範囲, ターゲットID)	予位から範囲内の距離にあるオブジェクトの中で予位に最も近いオブジェクトのIDをターゲットIDに入れる。
現在位置()	ポインタの現在位置を返す。
現在時刻()	システムの現在時刻を返す。
クリック位置()	直前にクリックされた位置を返す
速度(位置A, 位置B, 時刻A, 時刻B)	位置A, 位置B, 時刻A, 時刻Bからポインタの速度を求めて返す。
距離(位置A, 位置B)	位置Aと位置B間の距離を求めて返す。
予測位置(ピ位, X距, Y距, Z距)	ピ位, X距, Y距, Z距から予測位置を求めて返す。
X速度(速度, 方向)	速度, 方向から速度のX成分を求めて返す。
Y速度(速度, 方向)	速度, 方向から速度のY成分を求めて返す。
Z速度(速度, 方向)	速度, 方向から速度のZ成分を求めて返す。
予測距離(速度)	速度からターゲットまでの予測距離を求めて返す。
予測式生成(各ピ速, 各ピ位, 各ピ方, 各位置)	各ピ速, 各ピ位, 各ピ方, 各位置から回帰分析を行い, ピーク速度からターゲットまでの距離を予測する式を生成する。

10

20

【産業上の利用可能性】

【0045】

本発明は、ピーク速度からターゲット位置を予測することを利用するインタフェースを提供し、情報処理装置画面の大画面化に伴うマウス等のポインティング操作システムや、バーチャルリアリティ(VR)上の3次元内の物体を操作するデバイス、3次元トラックャー、アイソトニックなデバイスなどに利用できる。

30

【図面の簡単な説明】

【0046】

【図1】本発明に係るポインティング操作システムのターゲット位置予測アルゴリズムを示している。

【図2】ピーク速度検出処理のフローチャートの一例を示す。

【図3】ターゲット位置予測処理のフローチャートの一例を示す。

【図4】オブジェクト推定処理のフローチャートの一例を示す。

40

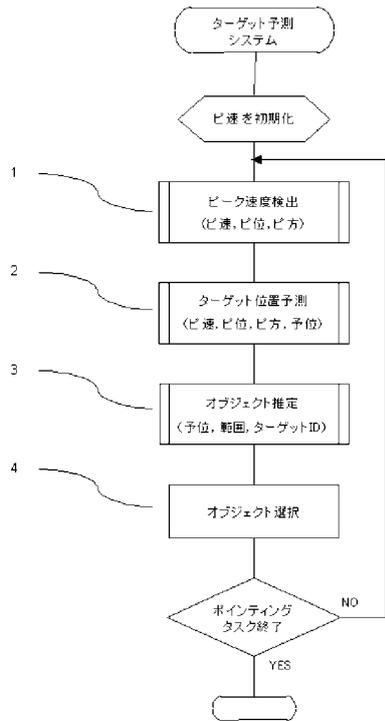
【図5】ポインティング操作システムのキャリブレーション処理のフローチャートを示す。

【符号の説明】

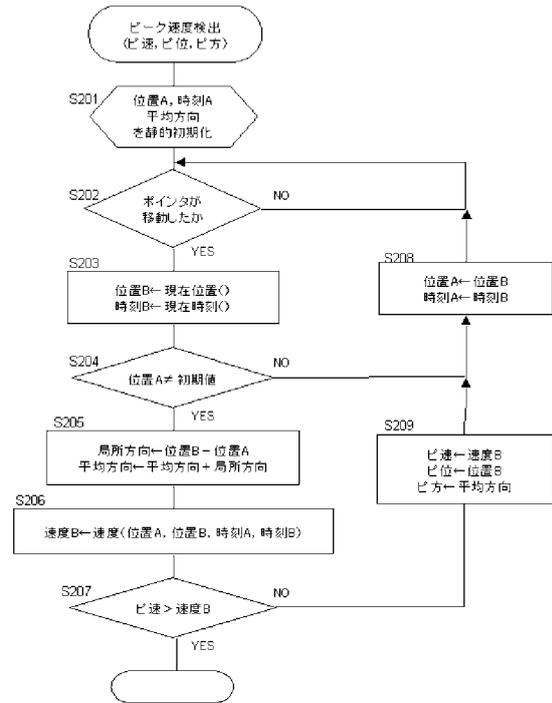
【0047】

- 1 ポインタのピーク速度検出処理
- 2 ターゲット位置予測処理
- 3 オブジェクト推定処理
- 4 オブジェクト選択処理

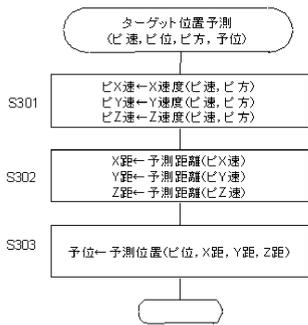
【 図 1 】



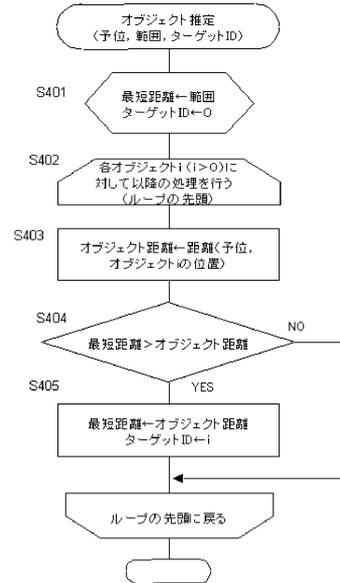
【 図 2 】



【 図 3 】



【 図 4 】



【 図 5 】

