

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号
特許第4938617号
(P4938617)

(45) 発行日 平成24年5月23日(2012.5.23)

(24) 登録日 平成24年3月2日(2012.3.2)

(51) Int.Cl.

F I

G O 6 F 3/033 (2006.01)

G O 6 F 3/048 (2006.01)

G O 6 F 3/033 4 2 2

G O 6 F 3/048 6 3 0

請求項の数 15 (全 21 頁)

(21) 出願番号	特願2007-271491 (P2007-271491)	(73) 特許権者	510276892
(22) 出願日	平成19年10月18日 (2007.10.18)		村井 幸輝郎
(65) 公開番号	特開2009-99042 (P2009-99042A)		奈良県奈良市北室町18-1 ファミール
(43) 公開日	平成21年5月7日 (2009.5.7)		奈良北室町305
審査請求日	平成22年10月18日 (2010.10.18)	(74) 代理人	100101454
			弁理士 山田 卓二
		(74) 代理人	100081422
			弁理士 田中 光雄
		(74) 代理人	100125874
			弁理士 川端 純市
		(72) 発明者	村井 幸輝郎
			奈良県奈良市北室町18-1 ファミール
			奈良北室町305
		審査官	中田 剛史
			最終頁に続く

(54) 【発明の名称】 オブジェクト操作装置、及び、デジタル画像フレームデータからマーカを特定する方法

(57) 【特許請求の範囲】

【請求項1】

撮像した操作体に基づき画面中のオブジェクトを操作するオブジェクト操作装置であって、

演算装置と、
上記演算装置と接続する表示装置と、
上記演算装置と接続する撮像装置と
を備え、

上記撮像装置は所定の操作体を撮像し、上記演算装置は撮像された上記操作体の画像を
マーカとして上記表示装置に表示し、

上記マーカの動きにより上記表示装置の画面におけるオブジェクトが操作されるオブジ
ェクト操作装置において、

上記演算装置は、
上記撮像装置が撮像したデジタル画像フレームデータから、所定の色相、彩度若しくは
明度を含む画像の持つ特性に着目して画素単位でビットマップB(x,y)を作成し、

上記ビットマップB(x,y)の、各々のy値におけるビットの存在を、始点(x)及
び幅(w)による線分要素(x,y,w)で表し、

上記各線分要素の間の接触関係の存否を判断し、接触関係にある線分要素を一まとめに
して物体要素とし、

上記物体要素から上記マーカを作成する

ことを特徴とするオブジェクト操作装置。

【請求項 2】

上記演算装置は上記操作体の画像の鏡像をマーカとして上記表示装置に表示することを特徴とする請求項 1 に記載のオブジェクト操作装置。

【請求項 3】

上記撮像装置は複数の操作体を撮像し、上記複数の操作体の各々に対応したマーカを上記オブジェクトに作用させることでオブジェクトが操作されることを特徴とする請求項 2 に記載のオブジェクト操作装置。

【請求項 4】

上記オブジェクトがウインドウであり、
複数の上記操作体による複数の上記マーカを上記ウインドウの内側で互いに接近させると、
上記ウインドウが表示画面の前面にないならば上記ウインドウが前面に移動し、上記ウインドウが表示画面の前面にあるならば上記ウインドウがドラッグされることを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 5】

上記オブジェクトがウインドウであり、
複数の上記操作体による複数の上記マーカを上記ウインドウ外側から上記ウインドウの一辺に接近させると、
上記ウインドウが表示画面の座標軸に平行に移動することを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 6】

上記オブジェクトがウインドウであり、
複数の上記操作体による複数の上記マーカのうち、1 個以上ずつのマーカを上記ウインドウの一組の対辺の夫々の外側から中心に近づけさせると、
上記ウインドウが、上記一組の対辺の夫々に交差し且つ中心に向かう方向に、縮小することを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 7】

上記オブジェクトがウインドウであり、
複数の上記操作体による複数の上記マーカのうち、1 個以上ずつのマーカを上記ウインドウの一組の対辺の夫々の内側から、対辺の夫々に接して広げさせると、
上記ウインドウが、上記一組の対辺の夫々に交差し且つ中心から遠ざかる方向に、広がることを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 8】

上記オブジェクトがウインドウであり、
3 個以上の上記操作体による 3 個以上の上記マーカのうち、1 個以上ずつのマーカを上記ウインドウの 3 乃至 4 辺の夫々の外側に接して存在させ且つそれらマーカを同じ速度ベクトルで移動させると、
上記ウインドウが平行に移動することを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 9】

上記オブジェクトがウインドウであり、
3 個以上の上記操作体による 3 個以上の上記マーカのうち、1 個以上ずつのマーカを上記ウインドウの 3 乃至 4 辺の夫々の内側に接して存在させ且つそれらマーカを中心から遠ざかる方向へ移動させると、
マーカの動きに合わせて上記ウインドウの幅及び高さが広がることを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 10】

上記演算装置において、マーカとオブジェクトの関係が操作内容毎に予め定義され、更に、複数の上記操作内容が基本的操作から複雑な操作へと予め順序付けされており、

10

20

30

40

50

更に、上記演算装置におけるフレームデータの処理において、上記順序に沿って、マーカとオブジェクトとが予め定義された上記関係を満たすか否かが確認され、満たさないことが確認されたときの上記順序付けにおける項数を“ k ”と置くと、 $(k - 1)$ 番目の上記順序付けに対応する操作内容がマーカからオブジェクトに対する操作となることを特徴とする請求項 3 に記載のオブジェクト操作装置。

【請求項 1 1】

上記演算装置におけるフレームデータの処理において、上記表示装置の画面上の複数のオブジェクトは、画面の手前から画面の奥へと順番付けされ、

マーカの操作の対象がどのオブジェクトであるかが、上記順番付けに沿って確認されることを特徴とする請求項 1 0 に記載のオブジェクト操作装置。

10

【請求項 1 2】

上記表示装置の画面上の最も手前にあるオブジェクト以外のオブジェクトに対して為される操作の種類が、上記表示装置の画面上の最も手前にあるオブジェクトに対して為される操作の種類よりも少ないことを特徴とする請求項 1 1 に記載のオブジェクト操作装置。

【請求項 1 3】

デジタル画像フレームデータにて、所定の色相、彩度若しくは明度を含む画像の持つ特性に着目して画素単位でビットマップ $B(x, y)$ を作成する工程と、

各々の y 値におけるビットの存在を、始点 (x) 及び幅 (w) による線分要素 (x, y, w) で表す工程と、

上記各線分要素の間の接触関係の存否を判断し、接触関係にある線分要素を一まとめにして物体要素とする工程と、

物体要素のうちからノイズを除去する工程と

を含む、デジタル画像フレームデータからマーカを特定する方法。

20

【請求項 1 4】

ビットマップ $B(x, y)$ を作成する工程が、

デジタル画像フレームデータの画素ごとに RGB 値から HSV 値に変換して H (Hue ; 色相)、 S (Saturation ; 彩度) 及び V (Value ; 明度) の値を取得する工程と、

予め設定されている HSV 値 (H_0, S_0, V_0) と、個々の画素 (x, y) での上記 HSV 値を、 H 、 S 、 V 夫々について比較し、夫々の差異が夫々の閾値以下の画素のみマーカ候補部分として抜き出してビットをオンとし、それ以外の画素ではビットをオフとする工程と

を含むことを特徴とする請求項 1 3 に記載の方法。

30

【請求項 1 5】

上記物体要素のうちからノイズを除去する工程が、

物体要素の面積が、第 1 の所定値より大きい値のもの、又は第 2 の所定値より小さい値のものを除去する工程と、

正円率が所定値より低い物体要素を除去する工程と、

数珠状のノイズを除去する工程と、

現在のフレーム上の各物体要素と、記録されている一つ前のフレーム上の各物体要素を対比させて、重心座標が同じ又は重心座標の差異が所定値以下である物体要素同士を、同じ物体要素として結び付ける工程と、

40

直前の一つ又は数枚のフレームと現在のフレームとにおける同一物体要素の重心座標の差異から、各物体要素の速度ベクトルを求める工程と

を含むことを特徴とする請求項 1 3 に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、コンピュータにおけるオブジェクト操作装置、及び、デジタル画像フレームデータからマーカを特定する方法に関する。

50

【背景技術】

【0002】

コンピュータのGUIをサポートする入力デバイスとして、現在、マウスが広く用いられている。しかしながら、マウスは例えば手や腕に障害を持つ人にとって、必ずしも有用な入力デバイスとは言えない。

【0003】

また、コンピュータ画像上で、マウスは必ず一点に関する入力動作を行う。しかしながら、コンピュータ画像に対して複数の点を介して同時に入力動作を行う入力デバイスは、現在見当たらない。即ち、コンピュータ操作者が複数の指で画面上の対象物にアプローチするような動作をサポートする入力デバイスは殆ど開発されていないといえる。

10

【0004】

なお、本願の先行技術として以下のようなものが挙げられるが、いずれも、コンピュータ操作者が複数の指で画面上の対象物にアプローチするような動作をサポートする入力デバイスを開示するものではない。

【特許文献1】特開2006-277666号公報

【特許文献2】特開2005-301693号公報

【特許文献3】特開2004-078488号公報

【特許文献4】特開2004-038429号公報

【特許文献5】特開2000-132340号公報

【特許文献6】特開2000-105671号公報

20

【発明の開示】

【発明が解決しようとする課題】

【0005】

本発明は、マウス以外の指や口によるコンピュータへの入力をサポートする入力装置を提供することを目的とする。

【課題を解決するための手段】

【0006】

本発明は上記の目的を達成するためになされたものである。本発明に係るオブジェクト操作装置は、

撮像した操作体に基づき画面中のオブジェクトを操作するオブジェクト操作装置であって、

30

演算装置と、

上記演算装置と接続する表示装置と、

上記演算装置と接続する撮像装置と

を備え、

上記撮像装置は所定の操作体を撮像し、上記演算装置は撮像された上記操作体の画像をマーカとして上記表示装置に表示し、

上記マーカの動きにより上記表示装置の画面におけるオブジェクトが操作されるオブジェクト操作装置である。そのようなオブジェクト操作装置において、

上記演算装置は、

40

上記撮像装置が撮像したデジタル画像フレームデータから、所定の色相、彩度若しくは明度を含む画像の持つ特性に着目して画素単位でビットマップ $B(x, y)$ を作成し、

上記ビットマップ $B(x, y)$ の、各々の y 値におけるビットの存在を、始点 (x) 及び幅 (w) による線分要素 (x, y, w) で表し、

上記各線分要素の間の接触関係の存否を判断し、接触関係にある線分要素を一まとめにして物体要素とし、

上記物体要素から上記マーカを作成する

ことを特徴とする。

【0007】

また、本発明に係るデジタル画像フレームデータからマーカを特定する方法は、

50

デジタル画像フレームデータにて、所定の色相、彩度若しくは明度を含む画像の持つ特性に着目して画素単位でビットマップB (x , y) を作成する工程と、

各々の y 値におけるビットの存在を、始点 (x) 及び幅 (w) による線分要素 (x , y , w) で表す工程と、

上記各線分要素の間の接触関係の存否を判断し、接触関係にある線分要素を一まとめにして物体要素とする工程と、

物体要素のうちからノイズを除去する工程とを含む。

【発明の効果】

【 0 0 0 8 】

本発明を利用することにより、表示画面に対する指や口などの相対的な動作によって、直接的にコンピュータへの入力操作を行うことができる。

【発明を実施するための最良の形態】

【 0 0 0 9 】

以下、図面を参照して本発明に係る好適な実施形態を説明する。

【 0 0 1 0 】

[[オブジェクト操作装置]]

図 1 は、本発明の実施形態に係るオブジェクト操作装置 2 の概略の機能を示すブロック図である。図 2 にも示すように、本発明に係るオブジェクト操作装置 2 は、表示装置 4 及び演算装置 6 を備えるコンピュータ 8、該コンピュータ 8 に接続するデジタル撮像装置 (D V カメラ) 1 0、及び、シール 1 2 で構成される。演算装置 6 の各機能部であるマーカ情報取得部 1 4、マーカ作成部 1 6、マーカ表示部 1 8、及び対象オブジェクト操作部 2 0 は、C P U が適切なコンピュータプログラムを実行することで実現される。デジタル撮像装置 (D V カメラ) 1 0 は、コンピュータ 8 の表示装置 4 の上端に組み込まれるものであってもよい (図 2 参照) 。

【 0 0 1 1 】

デジタル撮像装置 (D V カメラ) 1 0 は、図 3 に示すような指先に貼付された操作体としてのシール 1 2、及びその周辺や背景を撮像する。図 3 では、親指の先端部と人差し指の先端部とにシール 1 2 が 2 枚貼付されているが、その他の指の先端部に付されてもよい。また、指先ではなく、手のひらなどにシール 1 2 が付されてもよい。

【 0 0 1 2 】

なお、デジタル撮像装置 (D V カメラ) 1 0 は、ごく短い単位時間 (例えば、1 / 6 0 秒) 毎にフレームの形態で画像データを作成する。従って、オブジェクト操作装置 2 に含まれるマーカ情報取得部 1 4、マーカ作成部 1 6、マーカ表示部 1 8、及び対象オブジェクト操作部 2 0 は、フレーム単位で画像データを処理する。

【 0 0 1 3 】

シール 1 2 は、指先を全部又は一部覆う大きさであればよく、どのような形状であってもよいが円に近い形状であるのが好ましい。更に、シール 1 2 表面には所定の色、好ましくは一つの色 (例えば、黄色) が付けられている。

【 0 0 1 4 】

マーカ情報取得部 1 4 は、シール 1 2 の画像・位置の情報を含む画像情報をデジタル撮像装置 1 0 から取得し、マーカ作成部 1 6 に画像データを渡す。これを受けて、マーカ作成部 1 6 は、画像データの中から、シールの画像データを特定して抽出しこれをマーカ 2 2 とする。ここでマーカ 2 2 とは、コンピュータ画面上の対象 (オブジェクト) に対して操作を行う手段であり、通常マウス及びその (表示画面上の) ポインタが果たす役割の代わりと成り得る手段である。マーカ 2 2 は、コンピュータ 8 に実装されたソフトウェア上ではオブジェクトの一つとして扱われる。

【 0 0 1 5 】

マーカ作成部 1 6 は、形状及び大きさの特定された明確なマーカ 2 2 を作成することが望ましい。マーカ 2 2 の作成の具体例については、後で説明する。

10

20

30

40

50

【 0 0 1 6 】

マーカ作成部 1 6 により作成されるマーカ 2 2 は、マーカ表示部 1 8 によりコンピュータの表示装置 4 に表示される。このときマーカ表示部 1 8 は、撮像した画像データの鏡像を表示装置 4 に表示させる。そうすると、図 2 に示すような DV カメラ 1 0 により撮像されたシール 1 2 が、鏡像となってマーカ 2 2 として表示される。つまり、図 2 における親指に貼られたシール 1 2 (図では (A)) を付している) は、表示画面の左側のマーカとして表示され、人差し指に貼られたシール 1 2 (図では (B)) を付している) は、表示画面の右側のマーカとして表示される。

【 0 0 1 7 】

これらマーカ 2 2 が、図 2 における表示画面上のオブジェクト (図 2 ではウインドウ 2 4) に作用を及ぼすことにより、表示画面上のウインドウ (オブジェクト) 2 4 が操作される。即ち、操作者は、表示画面上のマーカ 2 2 を見て確認しつつ、自らの指や手の動きによりウインドウ 2 4 を操作することができる。

【 0 0 1 8 】

図 4 は、鏡像たるマーカ 2 2 の画像 (図 4 (1)) が、ウインドウ 2 4 を表示する画面 (図 4 (2)) に重ねられて、マーカ 2 2 とウインドウ (オブジェクト) 2 4 とを表示する画面となる様子を模式的に示している。

【 0 0 1 9 】

マーカ 2 2 がウインドウ (オブジェクト) 2 4 に作用を及ぼすことにより、ウインドウ 2 4 を操作する機能は、対象オブジェクト操作部 2 0 が行う。例えば、図 5 (2) に示すように、2 つのマーカ 2 2 がウインドウ 2 4 の右端に接してその右側にある場合 2 つのマーカ 2 2 が左に動けば (即ち、シールを貼った操作者の指が左に動けば) 右端を含むウインドウ 2 4 全体が右から左へ動くように、対象オブジェクト操作部 2 0 は、ウインドウ 2 4 の位置や形状を変化させる制御を行う。

【 0 0 2 0 】

[[対象オブジェクト操作部]]

次に、対象オブジェクト操作部 2 0 の動作を説明する。対象オブジェクト操作部 2 0 の動作に係るコンピュータプログラムでは、

- (1) マーカ、
- (2) レシーバ、
- (3) テンプレート、
- (4) オペレーション

が定義されて利用される。なお、図 1 1 は、本明細書におけるマーカ、レシーバ、テンプレート及びオペレーションの表記法を示す一覧表である。この表記法はコンピュータプログラムでの表現を基にしている。

【 0 0 2 1 】

マーカ 2 2 は、コンピュータ画面上の対象 (オブジェクト) に対して操作を行う手段であり、形 (a)、面積 (s)、速度ベクトル (v)、重心 (g) などをパラメータとして保持するオブジェクトである。その他のパラメータを含んでもよい。n 個のマーカの集合は例えば {M_n} で表現し、n 個目のマーカは M_n と表現する。n 番目のマーカの面積は M_n s と表現し、速度ベクトルの x 成分は M_n v.x と表現する。速度ベクトル (v) は、後でも説明するように、直前の一つ又は数枚のフレームと現在のフレームとにおける同一マーカの重心座標の差異から求められる。

【 0 0 2 2 】

なお、マーカ作成部 1 6、マーカ表示部 1 8 及び表示装置 4 は、マーカ 2 2 を表示するにあたり、マーカの撮像そのものを表示するのではなく、マーカ 2 2 の重心を中心とする所定半径の円 (例えば、半透明の円) を表示するようにしてもよい。

【 0 0 2 3 】

レシーバとは、コンピュータ画面上での操作の対象になるオブジェクトであり、本明細書では具体的な例としてウインドウ 2 4 を取り上げる。ウインドウ 2 4 は、左下座標 (x

10

20

30

40

50

、 y) と大きさ (w , h) を保持するオブジェクトである (w は幅であり、 h は高さである) (図 4 (2) 参照) 。更にレシーバは、直前フレームで当該レシーバになされた操作 (オペレーション) を記録するパラメータ (p) も保持する。例えば、直前フレームで当該レシーバになされた操作が、右から押す操作であったならば、 p は、

$p = k A c t i o n P u s h F r o m R i g h t$

等の定数となる。この p は、(後で詳述する) ノイズが多い場合などに、発行すべきオペレーションを決定するパラメータとなり得る。例えば、ノイズが多い場合には直前の p と現在の p が一致するときのみ、その p が示すオペレーションが発行される、というように利用される。

n 個のレシーバの集合は $\{ R_n \}$ で表現し、 n 個目のマーカは R_n と表現する。表示画面において、 R_n は R_{n+1} より必ず手前にある (即ち、 R_n は R_{n+1} の上に重なっている) ものとする。また、 n 番目のレシーバの左下 x 座標は $R_n \ x$ と表現する。

【 0 0 2 4 】

テンプレートとは、コンピュータ画面上での操作のテンプレートであり、特にレシーバを押す、握る、つかむというような操作の、一つ一つの条件を表す。テンプレートは次数 d と条件 Q を保持し、 $T_n = (d , Q (r , m))$ で表現される。ここで、次数 (d) とは、レシーバの幾つの辺に関わっているかを表す数であり (下記表 1 参照) 、例えば、一つの辺を外部から押す操作ならば “ 1 ” (図 5 (2) 参照) 、左右の辺で挟む操作ならば “ 2 ” (図 5 (3) 参照) 、外側をつかむ操作ならば “ 3 ” 又は “ 4 ” (図 5 (5) 参照) 、レシーバの内側をつかむ操作ならば “ 0 ” となる (図 5 (1) 参照) 。

$Q (R_k , \{ M_n \})$ は、一つのレシーバ R_k と、マーカの集合 $\{ M_n \}$ とを引数とする関数である。この関数 Q は、 $\{ M_n \}$ が R_k に対するその操作を表している場合には “ $T r u e$ ” (真) を、表していない場合には “ $F a l s e$ ” (偽) を、返す。

テンプレートは複数設けられ、 $\{ T_n \}$ で表現される。また、常に、

$T_n \ d \ T_{n+1} \ d$

が成り立つように、コンパイル時に $\{ T_n \}$ の内容の順番 (適用の優先順位) が項数として決定される。表 1 は、テンプレートの内容である。表の上から下へ $\{ T_n \}$ の n (項数) が順に大きくなる。なお、表 1 の内容や n (項数) の順序は、一つの例に過ぎないが、単純な動作から複雑な動作へと移行するように並べられている。

【 0 0 2 5 】

【 表 1 】

n	テンプレート	次数	内容
1	内側をつかむ	0	複数のマーカが、レシーバの内側で互いに接近する。
2	押す	1	複数のマーカが、外側からレシーバの一边に近づく。
3	挟む	2	1 個以上ずつのマーカが対辺の外側から中心に近づく。
4	広げる	2 - 4	1 個以上ずつのマーカが対辺の内側から辺に対して垂直に広がる。
5	外側をつかむ	3 - 4	1 個以上ずつのマーカが 3 乃至 4 辺の外側にあり、ほぼ同じ速度ベクトルで移動している。
6	内側から押し広げる	3 - 4	1 個以上ずつのマーカが 3 乃至 4 辺の内側にあり、全て中心から遠ざかる方向に移動している。

【 0 0 2 6 】

オペレーションとは、コンピュータ画面上での操作である。本発明では特に、レシーバを押す、握る、つかむというような操作の一つ一つを表す (下記表 2 参照) 。一つのオペ

レーション、例えば押すオペレーションには、押すテンプレートがあり、両者は対応している。本発明に係るオペレーションはプログラミングでいうところの関数であり、数列 $\{O_n\}$ で表される。また、 $O_n = (O(r, m))$ である。例えば、

$O_n = O(R_p, \{M_q\})$

は、マーカの集合 $\{M_q\}$ が n 番目のオペレーション O_n を引き起こし、そのオペレーション O_n を一個のレシーバ R_p に適用する、という意味である。

【0027】

【表2】

オペレーション	操作	図との対応
内側をつかむ	レシーバを前面に移動する。レシーバが前面にあるならばドラッグを開始する。	図5 (1)
押す	レシーバを座標軸に平行に移動する。	図5 (2)
挟む	レシーバの幅、又は高さを縮小する。	図5 (3)
広げる	レシーバの幅、又は高さを広げる。	図5 (4)
外側をつかむ	マーカの速度の平均ベクトルでレシーバを平行移動する。	図5 (5)
内側から押し広げる	マーカの動きに合わせて幅と高さを広げる。	図5 (6)

【0028】

オペレーションは、以下のようなものである。

【0029】

まず図5 (1) に示すように、複数のシールによる複数のマーカ22をウインドウ24の内側で互いに接近させる場合、ウインドウ24が表示画面の前面にないならばウインドウが前面に移動し (図5 (1 - 2) 参照)、ウインドウが表示画面の前面にあるならばウインドウがドラッグされる。

【0030】

次に、図5 (2) に示すように、複数のシールによる複数のマーカ22をウインドウ24外側からウインドウ24の一辺に接近させると、ウインドウ24が表示画面の座標軸に平行に移動する (図5 (2 - 2) 参照)。

【0031】

次に、図5 (3) に示すように、複数のシールによる複数のマーカ22のうち、1個以上ずつのマーカ22をウインドウ24の一組の対辺の夫々の外側から中心に近づけさせると、ウインドウ24が、その一組の対辺の夫々に交差し且つ中心に向かう方向に縮小する (図5 (3 - 2) 参照)。

【0032】

次に、図5 (4) に示すように、複数のシールによる複数のマーカ22のうち、1個以上ずつのマーカ22をウインドウ24の一組の対辺の夫々の内側から、対辺の夫々に接して広げさせると、ウインドウ24が、その一組の対辺の夫々に交差し且つ中心から遠ざかる方向に広がる (図5 (4 - 2) 参照)。

【0033】

次に、図5 (5) に示すように、3個以上のシールによる3個以上のマーカ22のうち、1個以上ずつのマーカ22をウインドウ24の3乃至4辺の夫々の外側に接して存在させ且つそれらマーカ22を同じ速度ベクトルで移動させると、ウインドウ24が平行に移動する (図5 (5 - 2) 参照)。

【0034】

更に、図5 (6) に示すように、3個以上のシールによる3個以上のマーカ22のうち、1個以上ずつのマーカ22をウインドウ24の3乃至4辺の夫々の内側に接して存在させ且つそれらマーカ22を中心から遠ざかる方向へ移動させると、マーカ22の動きに合

10

20

30

40

50

わせてウインドウ 24 の幅及び高さが広がる（図 5（6 - 2）参照）。

【0035】

ここで、マーカからウインドウの辺への作用について、図 6 を用いて説明する。図 6（1）の辺 AB に対しウインドウの外側からマーカ 22 が作用する場合を取り上げる。まず、辺 AB に関して長方形領域 p q r s を設定する。このとき、q、p が辺 AB を所定の比率（例えば、1 : 5 : 1）で内分し、q r 及び p s が所定の長さ（例えば 5）となるように、点 p、q、r 及び s を設ける。次に、マーカ 22 の重心が、その長方形領域 p q r s に包含されるか、が判定される。マーカ 22 の重心が長方形領域 p q r s に包含される場合、マーカ 22 の速度ベクトルの、マーカ 22 の重心から辺 AB に直交する半直線に関する \cos が求められ、 \cos から θ が求められる。 θ が所定の範囲（例えば、 -40° 以上 $+40^\circ$ 以下）にあれば、マーカ 22 が辺 AB に外側から作用しているものとする。このとき、マーカ 22 は、速度ベクトルの辺 AB に直交する成分（のみ）、つまり直交するベクトルへの正射影、で辺 AB を動かそうとしているとしてもよいし、速度ベクトルそのもので辺 AB を動かそうとしているとしてもよい。

10

【0036】

辺 AB に対しウインドウの内側からマーカ 22 が作用する場合には、辺 AB に関して長方形領域 t u q p を設定し、マーカ 22 重心がその長方形領域 t u q p に包含されるが判定され、更に、マーカ 22 の速度ベクトルとマーカ 22 重心から辺 AB に直交する半直線との角 θ が所定の範囲にあるか、が判定される。同様に、夫々の辺に対する作用では、長方形領域（S11、S12、S21、S22、S31、S32）を設定して、マーカ 22 の重心の包含が判定されることになる。なお、図 6（2）に示すように、ウインドウが近接する場合には、近接する辺の中線（図 6（2）では、vw）により、両方の長方形領域が設定されることになる。

20

【0037】

続いて、マーカ、レシーバ、テンプレート、及びオペレーションの具体的関係を説明する。

【0038】

後で説明するようにマーカの特定がなされると、マーカがレシーバに対してどのような操作（作用）を行うものであるかが特定される。このとき、レシーバとテンプレートとの照合が、行われる。

30

【0039】

まず、画面上で一番手前のレシーバ R_1 について、テンプレート T_1 からテンプレート T_n まで、

条件 $T_p = [1, n] \quad Q(R_1, \{Mq\})$

が、True を返すのか False を返すのかを調べる。最初に False が返されたときのテンプレートの項数を “k” と置くと、オペレーションとして、

$O_{k-1} \quad O(R_1, \{Mq\})$

が実行される。

【0040】

また、テンプレート T_1 とテンプレート T_2 とのいずれについても、

40

条件 $T_p = 1 \text{ and } 2 \quad Q(R_1, \{Mq\})$

が False を返す場合、対象レシーバは R_2 に移行する。つまり、より単純な動作であり、 T_3 以降のテンプレートの基本動作となる T_1 と T_2 のテンプレートの条件 Q が偽（False）を返す場合、 R_1 にはマーカは作用しないものとする。

【0041】

上記に続き、レシーバ R_2 について、

条件 $T_1 \quad Q(R_2, \{Mq\})$

が、True を返すのか False を返すのかを調べるのであるが、テンプレート T のうち、つかむ動作のみが照合の対象となる。表 1 では、「内側をつかむ」（ T_1 ）のみが照合の対象となる。ただし、一般的に、人間の行為は最も手前にあるもの（オブジェクト）

50

に集中するからであり、「押す」「挟む」「広げる」「内側から押し広げる」などの操作は R_1 （最も手前のレシーバ）のみが対象とされるべきだからである。また、内側を掴む動作の場合でも、それに関わるマーカがレシーバ R_2 の可視領域内（つまり、他のレシーバで隠れていない部分）に存在することが条件となる。

【0042】

なお、上記「つかむ」動作により、レシーバ R_2 は一番手前に移動される。一番手前に移動されれば、このオブジェクトは以降のフレームにおいて、「つかむ」動作以外の動作の対象と成り得る。

【0043】

R_2 について、「つかむ」動作のテンプレートの条件 Q がいずれも $False$ であれば、以下順次同様に、レシーバ R_p （ $p = 3$ ）についても、

$T_1 = Q(R_p, \{Mq\})$

が、 $True$ を返すのか $False$ を返すのかを調べるのであるが、テンプレート T のうち、つかむ動作のみが照合の対象となる。

【0044】

対象オブジェクト操作部20は、以上のようにフレーム単位で画像データを処理する。以降のフレームについても、同様に処理を行う。

【0045】

[[マーカ作成部]]

図7は、オブジェクト操作装置2におけるマーカ作成部16の、フレーム単位の画像データからマーカを抽出するまでの処理を表すフローチャート図である。また、図8は、図7に示すフローチャート図における「ノイズを除去する」工程（S08）を更に詳細化したフローチャート図である。図7及び図8を用いてマーカ作成部のマーカ抽出処理を説明する。

【0046】

まず、フレーム単位の画像データに対してメジアンフィルタをかける（工程S02）。メジアンフィルタは従来技術のものでよい。また、この工程S02は省略されてもよい。

【0047】

次に、画像データをビットマップ化する（工程S04）。この工程S04において、まず、画像データを画素単位でRGB値からHSV値に変換しH（Hue；色相）、S（Saturation；彩度）及びV（Value；明度）の値を取得する。予め設定されているマーカ色（例えば、黄色）の所定のHSV値（ H_0, S_0, V_0 ）と、個々の画素（ x, y ）でのHSV値を、H、S、V夫々について比較し、夫々の差異が夫々の閾値以下の画素のみマーカ候補部分として抜き出して、ビットマップを作成する。

【0048】

次に、マーカ候補ビットマップデータをエレメント指向データ化する（工程S06）。このエレメント指向については、後で説明する。

【0049】

最後に、ノイズの除去を行う（工程S08）。ノイズの除去により、マーカが明確化され且つマーカの速度が把握される。このノイズ除去処理の詳細については、後で説明する。

【0050】

[[エレメント指向]]

エレメント指向とは、 n 次元の弧状連結の立体を、互いに接している個々の（ $n - 1$ ）次元の立体で表すアルゴリズム、及びその背景にある考え方を言い、本願発明者が案出した考え方である。本明細書では、2次元の弧状連結の平面図形を、互いに接している個々の線分で表す、というエレメント指向の考え方を利用することになる。このエレメント指向は、例えば、多数の平行平面画像で構成される人体のCTスキンの画像から3次元の立体臓器のモデルを形成する場合や、コンピュータグラフィックスの3次元のモデリングツールを構築する場合などに、利用され得る。

10

20

30

40

50

【 0 0 5 1 】

なお、本明細書では、プログラミング言語としてC言語の表記を利用するが、他の言語であってもよい。図12は、本明細書におけるエレメント指向表記法を示す一覧表である。この表記法はコンピュータプログラムでの表現を基にしている。

【 0 0 5 2 】

まず、ビットマップを用意する。ビットマップでは、任意の点(画素)(x, y)における値がTrueかFalseか(即ち、オン“1”かオフ“0”か)である。 (x, y) におけるビットマップの値を、 $B(x, y)$ と表すことにする。

【 0 0 5 3 】

次に、 $B(x, y)$ の値から、線分要素を初期化する。以下のプログラム(表3)に示されるように、 $B(x, y)$ を連続的に観察する。まず、 y (縦軸)を固定して x (横軸)を連続的に増加させる。 $B(x, y) = \text{True}$ となる点が (x_0, y) から (x_1, y) まで分布しているとすると、そこに線分が成立している。この線分を線分要素とし、始点である (x_0, y) の座標と横幅 $x_1 - x_0 = w$ の値を、 $(x, y, w) = (x_0, y, x_1 - x_0)$ として保持する。ここで、線分要素が複数存在するものと考え、それら線分要素を、 $\{L_n\}$ という数列で表し、線分要素の始点座標と幅とを、夫々 $L_n \cdot x$ 、 $L_n \cdot w$ というように表すものとする。例えば、図9(2)には、 $L_1 \sim L_5$ が示されていることになる。

【 0 0 5 4 】

【表3】

```
int x, y;
for (y=0 ; y<(高さ);y++)
{
    for (x=0 ; x<(幅) ; x++)
    {
        BOOL b = B(x, y);
    }
}
```

【 0 0 5 5 】

$\{L_n\}$ の全てを頂点とする位相幾何学的な無向グラフ $G := \{g, \{L_n\}, E\}$ を考える。そして、 L_a と L_b とが代数幾何学的に接触していれば、 L_a と L_b との間に辺が存在すると考える。接触は以下のように判断する。

【 0 0 5 6 】

【表4】

```
int D = (La→(x+w) - Lb→x) × (La→x - Lb→(x+w))
```

(判断基準): $La \cdot y$ と $Lb \cdot y$ との差の絶対値が1のとき、上式のDが正ならば接触していない、Dが0ならば点で接触している、Dが負ならば線分で接触している。 $La \cdot y$ と $Lb \cdot y$ との差が0のとき、上式のDが0ならば接触している。それ以外の場合は接触していない。

【 0 0 5 7 】

上述に従い、線分要素の接触関係の存否を整理する(即ち、上述に従い必要に応じて辺を設定する)。できあがったグラフGにおいて、弧状連結な部分グラフの集合 $\{G'\}$ を物体要素の集合とし、 $\{G'\}$ の一項目 $G' = \{L_n\}$ 要素を物体要素とする。そうすると、個々の物体要素は、一つの弧状連結である形状を表すことになる(図9(2)参照)。

【 0 0 5 8 】

物体要素に様々なアルゴリズムを適応して、様々な数値を簡易に表現できる。以下では、面積、重心座標、正円からの変形率を示す。なお、以下の A_n は一つの物体要素である

。

【 0 0 5 9 】

面積は、以下の式で表される。

【 数 1 】

$$S = \sum_{k=1}^p An \rightarrow L_k \rightarrow w$$

ここで、pはAnを構成する線分要素の数である。上式は、線分要素を総和すれば面積となることを示しているに過ぎない。

【 0 0 6 0 】

10

重心座標は以下の式で表される。

【 数 2 】

$$(gx, gy) = \frac{\sum_{k=1}^p An \rightarrow L_k \rightarrow ((x + \frac{w}{2}) \times w, y \times w)}{S}$$

【 0 0 6 1 】

正円からの変形率は以下の式で表される。

【 数 3 】

$$x' = \min(An \rightarrow L_p \rightarrow x)$$

20

$$y' = \min(An \rightarrow L_p \rightarrow y)$$

$$w' = \max(An \rightarrow L_p \rightarrow (x + w)) - x'$$

$$h' = \max(An \rightarrow L_p \rightarrow y) - y'$$

$$c = \left| 1 - \frac{4S}{w' \times h' \times \pi} \right|$$

【 0 0 6 2 】

図9は、従来技術のラベリング処理とエレメント指向との対比を示す概念図である。図9(1)が、ラベリング処理による図形の表現の図であり、図9(2)が、本発明に係るエレメント指向データによる図形の表現の図である。図から明白なように、ラベリング処理に比べてエレメント指向では、データ量が大幅に減少する。

30

【 0 0 6 3 】

[[ノイズの除去]]

図8は、ノイズの除去の工程を詳細化したフローチャート図である。このノイズの除去の工程を、図10に示す画面の物体要素の例を用いて説明する。図10に示す物体要素の例では、3つの黄色いマーカ、黄色い筆記具、黄色い人形、及び、黄色の模様の入ったカーテンが含まれている。

【 0 0 6 4 】

最初に、物体要素の面積Mn sが、非常に大きい値のもの又は非常に小さい値のものを除去する(S0802)。例えば、1画素の物体要素や、画面の略半分を覆うような物体要素を、除去する。図10の例では、筆記具が除去され得る。

40

【 0 0 6 5 】

次に、正円率の低い物体要素を除去する(S0804)。物体要素の正円率は、上記[数3]の5つめの式の右辺第2項で表される。つまり、物体要素の形状が円に近い程、正円率は1に近くなり、従って、変形率c[物体要素における変形率M(n(c))]が小さくなる。そこで、変形率M(n(c))が所定値より大きいものを除去する。この除去により、例えば、とても変わった形状の物体要素や砂嵐状のノイズを示す物体要素が省かれる。図10に示す人形はこの工程で除去される。仮に、工程S0802にて、図10に示す筆記具が除去されなかったとしても、この工程で除去される。

50

【 0 0 6 6 】

次に、数珠状ノイズを除去する（S 0 8 0 6）。数珠状ノイズとは、図 1 0 に示す模様付きのカーテンのように、局所的に複数の動きのある物体要素が集中しておりそれら物体要素の面積が比較的等しいもののようなノイズである。数珠状ノイズの除去は、以下のように行う。

【 0 0 6 7 】

関数 $F(x, d)$ (x は物体要素、 d は正の数) が、ある物体要素 x から距離が d 以内である他の物体要素の集合を返す関数であるとする（但し、2 つの物体要素の距離は、両者の重心の距離である）。また、関数 $G(x, d)$ (x は物体要素、 d は正の数) は、ある物体要素 x に関して $\{y\} = F(x, d)$ を実行し、戻り値の集合が空集合でなければその集合の要素全て y に関して更に $\{z\} = F(y, d)$ を実行し、またその戻り値の集合の要素 z につき $F(z, d)$ を実行し、またその戻り値の要素につき・・・と再帰的に繰り返して空集合が返されるまで続け、空集合が返された時点でそれまでに戻り値に含まれていた物体要素全てを集合に入れて返す関数であるとする。

10

【 0 0 6 8 】

更に、関数 $H(x, d)$ (x は物体要素、 d は正の数) は、関数 $G(x, d)$ の戻り値の集合の要素の数を返す関数であるとする。

【 数 4 】

$$H'(x, d) = \Delta H(x, d) / \Delta d$$

$$H'(x, d) = \frac{\Delta H(x, d)}{\Delta d} = \frac{H(x, d + \Delta d) - H(x, d)}{\Delta d}$$

20

上式のように関数 $H'(x, d)$ を定義したとき、 $H'(x, d)$ の値が所定の閾値を超えているならば、 $G(x, d)$ にノイズが含まれている可能性が高いことになる。従って、 $G(x, d)$ の物体要素全てについて面積に関する偏差値を求め、その偏差値が 5 0 に近いものをノイズであると判断して、除去する。

【 0 0 6 9 】

上記の関数 F 、 G 、 H 、 H' による判断及び除去は、例えば、 $3 \times 3 = 9$ 分割の分割画面ごとになされてもよい。

【 0 0 7 0 】

30

この時点で残っている物体要素はマーカとすることができる。前に説明したように、マーカでは重心と速度（ベクトル）が重要であるから、上記【数 2】の式に従いマーカの重心を求め、更に速度ベクトルを求めておく。まずこの時点で、現フレーム上の各物体要素と、一つ前のフレーム上の各物体要素とを対比させて、同じ物体要素を結び付ける（S 0 8 0 8）。前後 2 つのフレーム間で同じ物体要素を結び付けるに当たっては、重心座標が同じ又は重心座標の差異が所定値以下である 2 つの物体要素を、同じ物体要素とすればよい。また、後続のフレームにおける処理のために、この時点での物体要素を含む現フレームのフレームデータを記録しておく。

【 0 0 7 1 】

次に、物体要素の速度ベクトル $M_n \quad v$ を算出する（S 0 8 1 0）。速度ベクトル $M_n \quad v$ は、直前の一つ又は数枚のフレームと現在のフレームとにおける同一物体要素の重心座標の差異から求められる。前にも説明したように、マーカを表示するにあたっては、マーカの撮像そのものを表示するのではなく、マーカの重心を中心とする所定半径の円（例えば、半透明の円）を表示するようにしてもよい。これに加えて、随時変化する速度ベクトルを表す矢印をマーカの重心から伸ばして示すようにしてもよい。

40

【 0 0 7 2 】

以上の工程 S 0 2 ~ 工程 S 0 8、及び、工程 S 0 8 0 2 ~ 工程 S 0 8 1 0 により、フレーム毎のマーカが作成される。

【 0 0 7 3 】

[変形例 1]

50

上記ノイズの除去の説明において、図8に示すフローチャートを用いてノイズの除去の工程を詳細に示したが、ノイズの除去の工程は上述のものに限定されるわけではない。図13に示すように、物体要素が他のサイクル状の物体要素内部にある状況を考える。なお、図13には示されていないが、内部の物体要素は「緑」の物体要素G (Green Area)、その外延にサイクル状に存在する物体要素は「青」の物体要素B (Blue Area)であるとする。

【0074】

このようにある物体要素が別の物体要素のサイクル内に存在する場合、サイクル状の物体要素はノイズであると考えられる。

【0075】

10

以下(表5)に、物体要素Gが別の物体要素Bのサイクル内に存在するの否かを判定する工程を述べる。判定する工程は、図8のフローチャートを構成する各工程間のどこで行われてもよいし、工程S0810の後に行われてもよい。

【0076】

[表5]

BOOL GreenAreaIsInsideBlueArea(AreaElement G, AreaElement B)

```
{
    int i, j;
    for (i=0 ; i<Gの線分要素の個数 ; i++)
    {
        int l[十分大きな整数];
        BOOL inside = NO;
        int x = G(i(x));           //Gのi番目の線分要素の始点のx座標
        int y = G(i(y));           //Gのi番目の線分要素の始点のy座標
        int w = G(i(w));           //Gのi番目の線分要素の幅
        int c = 0;                  //y座標が一致する線分要素の数
        for (j=0 ; j<Bの線分要素の個数 ; j++)
        {
            if (y == B(j(y)))      //y座標が一致するものがあれば
            {
                l[c] = j;
                c = c+1;            //一致しているから数を1プラス
            }
        }
        if (c > 1)
        {
            BOOL left_exists = NO; BOOL right_exists = NO;    //Gの線分要素
            の左、右にBの線分要素があるか
            int k;
            for (k=0 ; k<c ; k++) //y座標が同じものについてループ
            {
                LineElement L = B(l[k]);
                if (x + w < L(x))
                {
                    right_exists = YES;
                } else if (L(x) + L(w) < x) {
                    left_exists = YES;
                }
            }
            if (left_exists)

```

20

30

40

50

```

        {
            if (right_exists)
            {
                inside = YES;
            } else {
                inside = NO;
            }
        } else {
            inside = NO;
        }
    } else {
        inside = NO;
    }

    if (inside == NO)
    {
        return NO;          //左右ともに囲まれていない線分があれば
直ちに関数から抜け出してNOを返す
    }
}

return YES;          //一度もinsideがNOになっていないから、全ての線分要素が左右囲ま
                     //れているから、物体要素も左右囲まれている。よって、GはBの中にある。
}

```

【 0 0 7 7 】

ある物体要素が別の物体要素のサイクル内に存在することを数学的に表記すると、以下
のようになる。

【数 5】

$$A_q \rightarrow L_{\nabla r} \rightarrow y = A_p \rightarrow L_{\{Sn\}} \rightarrow y$$

$$A_q \rightarrow L_{\nabla r} \rightarrow x > A_p \rightarrow L_{St} \rightarrow (x + w)$$

$$A_q \rightarrow L_{\nabla r} \rightarrow (x + w) < A_p \rightarrow L_{St} \rightarrow x$$

つまり、上記 3 式全てを満たす有限数列 { S n }、自然数 t、自然数 u が存在し、 t u
であるとき、A q は A p の内部にあることになる。

【 0 0 7 8 】

なお、上記表 5 に示す、物体要素 G が別の物体要素 B のサイクル内に存在するの
かを判定する工程は、2 次元の物体要素 G、B に関するものである。エレメント指向は n 次
元 (n 3) の物体要素を対象にできる考え方であるから、ある n 次元の物体要素が別の
n 次元の物体要素のサイクル内に存在するか否かの判定にもエレメント指向の考え方を利用
できる。但し、n 次元 (n 3) の物体要素のサイクル関係の判定では、2 次元の物体
要素のサイクル関係の判定 (表 5 参照) よりも考慮すべき点が増大する。

【 0 0 7 9 】

このように、エレメント指向は、単に物体の代数幾何学的な不連続 (性) を表すのに適
するだけでなく、物体の位相幾何学的な特徴を表すのにも適する考え方であるといえる。

【 0 0 8 0 】

[変形例 2]

上記マーカ作成部の説明において、マーカ作成の一例を述べたが、マーカ及びシールは
上述のものに限定されるわけではない。例えば、親指には赤色のシールを、親指以外の指
には黄色のシールを貼り、親指以外の指のシールの鏡像たるマーカは、上述の「対象オブ
ジェクト操作部」の説明で述べたマーカと同じ機能を有し、親指の (赤色の) シールの鏡

像たるマーカ（の表示）は、マウスのクリックの機能を有するようにしてもよい。

【0081】

つまり、親指を曲げて親指の赤色シールがDVカメラに写らないようにした上で、他の4本の指でウインドウ24を操作し、途中で親指を伸ばして親指の赤色シールがDVカメラに写るようにする（即ち、赤色シールの鏡像たるマーカが突然表示される）とクリック操作が発生する。

【0082】

また、左手の親指には赤色のシールを、右手の5本の指には黄色のシールを貼り、右手の指のシールの鏡像たるマーカは、上述の「対象オブジェクト操作部」の説明で述べたマーカと同じ機能を有し、左手親指の（赤色の）シールの鏡像たるマーカの表示は、マウスのクリックの機能を有するものであってもよい。

10

【0083】

また、シールがストロー等の短い棒状物の先端に貼られ、操作者がそのストローをくわえ口によりシール及びマーカを動かすようにしてもよい。

【0084】

タッチスクリーンがマーカの入力源（マーカ情報取得部14）であってもよい。つまり、画面がタッチスクリーンを兼ねており、その画面への圧力を伴う接触による接触領域がマーカとなるようにしてもよい。

【図面の簡単な説明】

【0085】

20

【図1】本発明に係るオブジェクト操作装置の概略の機能を示すブロック図である。

【図2】本発明に係るオブジェクト操作装置における、シールとマーカの関係を示す図である。

【図3】指先にシールを付した様子を示す図である。

【図4】鏡像たるマーカの画像（図4（1））が、ウインドウを表示する画面（図4（2））に重ねられて、マーカとウインドウ（オブジェクト）とを表示する画面となる様子を模式的に示す図である。

【図5】マーカとウインドウとの関係を示す6つの図である。

【図6】マーカからウインドウの辺への作用を説明するための図である。

【図7】本発明に係るオブジェクト操作装置におけるマーカ作成部の、フレーム単位の画像データからマーカを抽出するまでの処理のフローチャート図である。

30

【図8】図7に示すフローチャート図における「ノイズを除去する」工程（S08）を更に詳細化したフローチャート図である。

【図9】従来技術のラベリング処理とエレメント指向との対比を示す概念図である。

【図10】画面上の物体要素の例を示す図である。

【図11】本明細書におけるマーカ、レシーバ、テンプレート及びオペレーションの表記法を示す一覧表である。

【図12】本明細書におけるエレメント指向表記法を示す一覧表である。

【図13】2つの物体要素の例を示す図である。

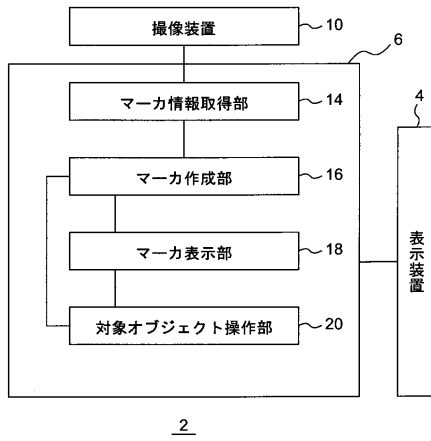
40

【符号の説明】

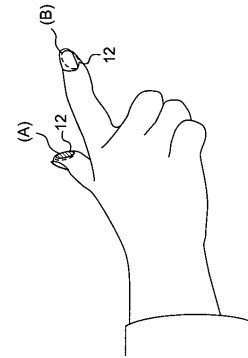
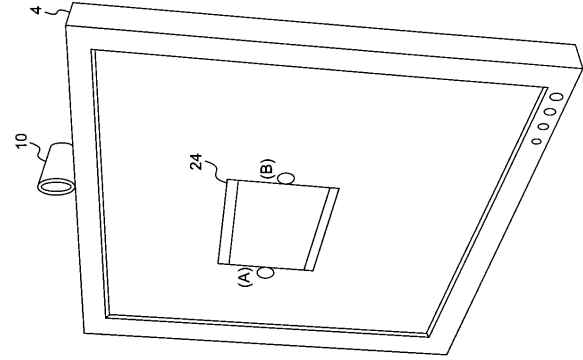
【0086】

2・・・オブジェクト操作装置、4・・・表示装置、8・・・コンピュータ、10・・・デジタル撮像装置（DVカメラ）、12・・・シール、14・・・マーカ情報取得部、16・・・マーカ作成部、18・・・マーカ表示部、20・・・対象オブジェクト操作部、22・・・マーカ、24・・・ウインドウ。

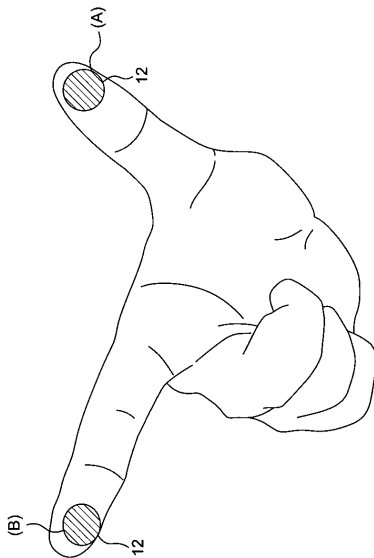
【図 1】



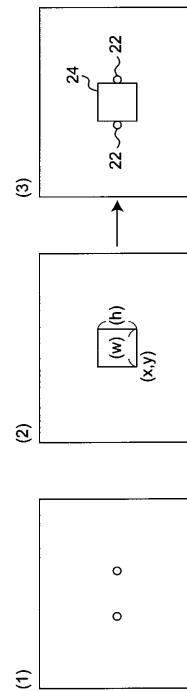
【図 2】



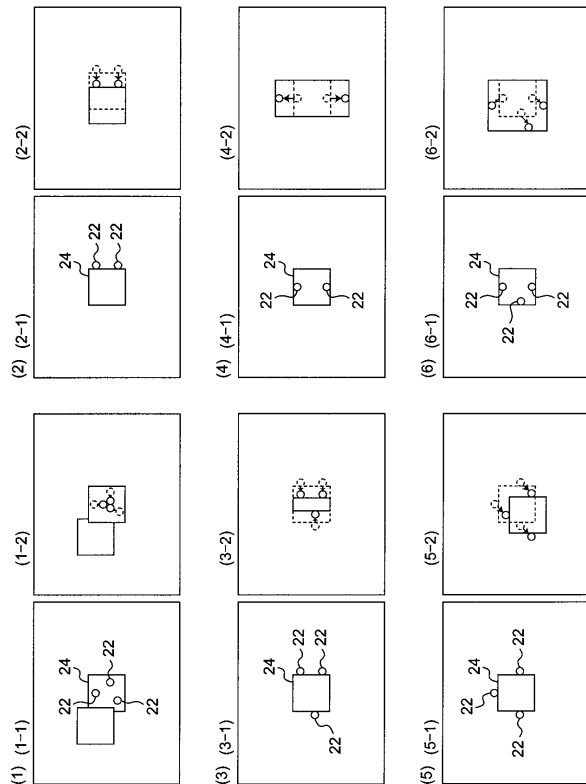
【図 3】



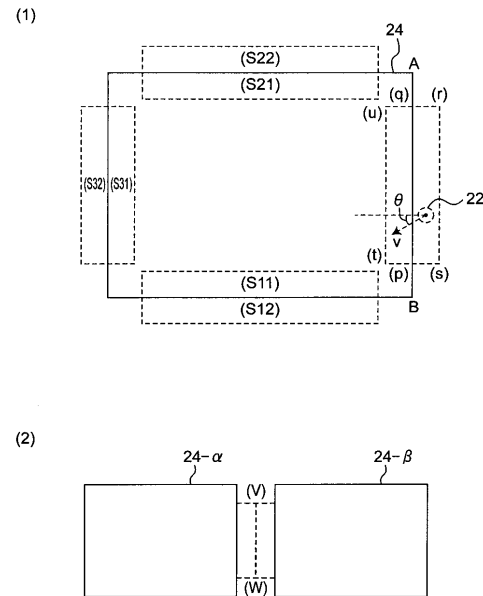
【図 4】



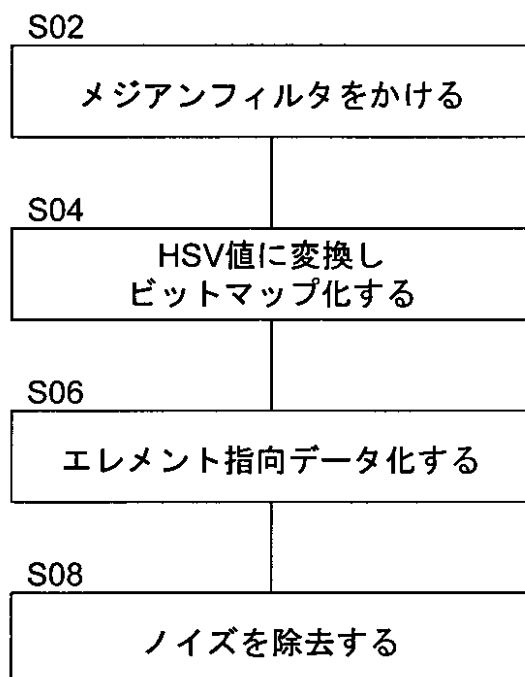
【図 5】



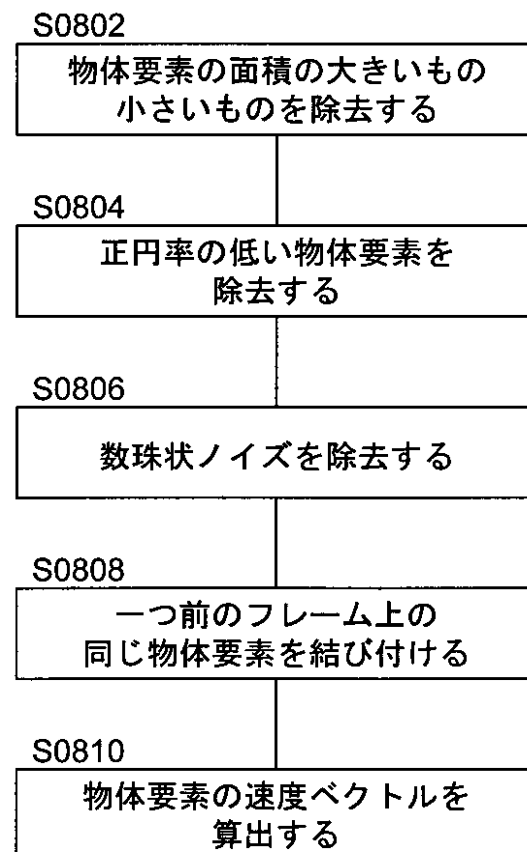
【図 6】



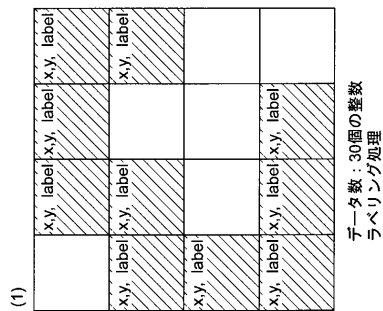
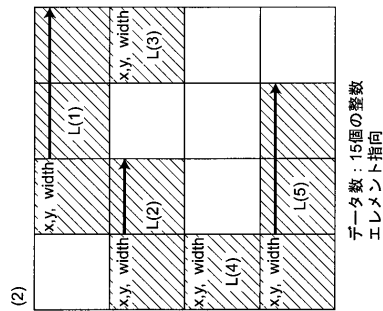
【図 7】



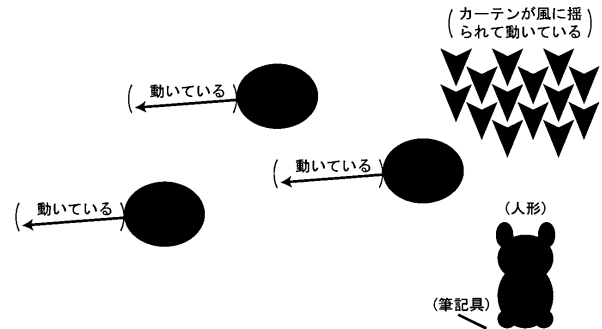
【図 8】



【図 9】



【図 10】



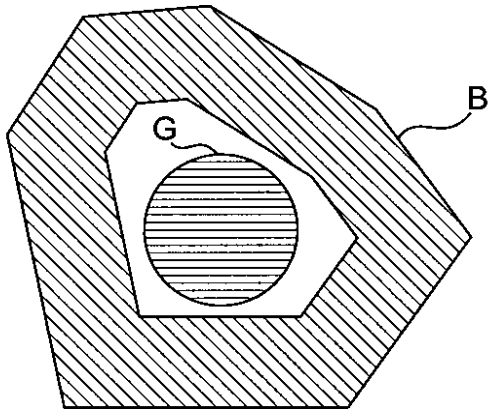
【図 11】

数学演算の表記	
和：A+B 差：A-B 実数のかけ算：A×B ベクトルの内積：A・B 関数：f(x, y, ...) ベクトルvの成分：v, x, y, v, y	
マーカの表記	
1つのマーカ：m = (a, s, v) (a：一つの物体要素、s：正の整数、v：XY平面上的速度ベクトル) n個目のマーカ：Mn = (a, s, v) n個のマーカの数列：{Mn} マーカのフィールド取得：m→a, m→s, m→v 又は、Mn→a, Mn→s, Mn→m 複数階層にまたがるフィールドの取得：Mp→a→Lq→xなど	
レシーバの表記	
1つのレシーバ：r = (x, y, w, h, p) (x, y, w, h：ウィンドウの範囲、p：直前のテンプレート) n個目のレシーバ：Rn = (x, y, w, h, p) n個のレシーバの数列：{Rn} レシーバのフィールド取得：r→x, r→y, r→w, r→h, r→p 又は、Rn→x, Rn→y, Rn→w, Rn→h, Rn→p	
テンプレートの表記	
1つのテンプレート：t = (d, Q(r, m)) (d：次数、Q：一つのレシーバとマーカの数列を引数にとる関数、r：引数のレシーバ一個、m：引数のマーカの数列) n個目のテンプレート：Tn = (d, Q(r, m)) n個のテンプレートの数列：{Tn} テンプレートのフィールド取得：t→d 又は、Tn→x テンプレートの関数実行：t→Q(Rp, {Mp}) 又は、Tn→Q(Rp, {Mp})	
オペレーションの表記	
1つのオペレーション：o = (0(r, m))：一つのレシーバとマーカの数列を引数にとる関数でこれを実行するとレシーバが変形する、r：引数のレシーバ一個、m：引数のマーカの数列) n個目のオペレーション：On = (0(r, m)) n個のオペレーションの数列：{On} オペレーションの関数実行：o→0(Rp, {Mp}) または、On→0(Rp, {Mp})	

【図 12】

数学演算の表記	
和：A+B 差：A-B 実数のかけ算：A×B ベクトルの内積：A・B 関数：f(x, y, ...) 位相幾何学的なグラフ：G := {g, V, E} (V, Eは数列、集合のどちらでも良い)	
エレメントの表記	
1つの線分要素：l = (x, y, w) n個目の線分要素：Ln = (x, y, w) n個の線分要素の数列：{Ln} 線分要素のフィールド取得：l→x, l→y, l→w 又は、Ln→x, Ln→y, Ln→w 1つの物体要素：a = {Ln} n個目の物体要素：An = {Ln} n個の物体要素の数列：{An} 物体要素のフィールド取得：a→{Ln} (構成する線分要素全てを取得)、a→Ln (構成するn個目の線分要素を取得) 又は、An→{Ln}、An→Ln 複数階層にまたがるフィールドの取得：Ap→Lq→xなど	
エレメント内の数学演算の省略法	
同・エレメント内のフィールド値の演算の場合、 Ln→x + Ln→w は、 Ln→(x+w) のように省略できる。+以外の演算子でも可能である。 別エレメント間1上のフィールド値の演算は、省略することは出来ない。	

【図 13】



フロントページの続き

(56)参考文献 特開2003-346162(JP,A)
特開2004-240672(JP,A)
特許第3953450(JP,B2)
特開平07-182101(JP,A)
特開2004-220419(JP,A)
国際公開第99/52072(WO,A1)
特開昭62-95686(JP,A)
特開2000-11171(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F	3/033
G06F	3/048